

---

# 의료이력 관리 프로그램 기획의도와 개발 코드

12204152 박희윤

## 1. 프로젝트 주제

---

환자 관리 프로그램

Patient management.

Patient care.

## 2. 프로젝트 개요

---

환자들의 정보를 자료화 하여 정리할 수 있는 프로그램이다.

여러 데이터를 취합하여 차트로 만들어 가독성을 높여 준다.

환자들에 연령, 병명, 진료이력 등을 기록하고 관련된 사항 등을 정리하여 자료화 시키고 필요할 때 마다 열람 한다.

환자들의 재방문을 고려해 정보들을 데이터화 하여 보다 효과적으로 업무처리 할 수 있게 도와준다.

### 3. 구현 기능(기능 소개)

---

- 환자의 성명, 생년월일, 전화번호, 병명, 진료 내역 기록
  - 환자의 담당의사, 담당 간호사 기록
  - 의사, 간호사 별 담당 환자들을 별도 검색
  - 월 별 접수된 환자의 수, 여러 종류의 병을 차트화
  - 생년월일로 연령을 파악하고 데이터 정렬
  - 환자 재방문 시 환자 이름을 검색하여 이전 기록 열람
  - 병원근로자 직원 관리 기능 추가
  - 원장님들을 위한 주가 차트 제공
- 등등

## 4. 기대효과

---

여러 병원에서 환자를 보다 수월하고 편리하게 관리 할 수 있으며 환자들의 정보, 진료이력, 과거병력 등을 화면상에 출력해준다.

전체적인 환자들에 연령, 병명 등을 데이터화 시켜 월 별 자료 정리가 용이하게 해준다.

병원에서 일하는 근로자들의 담당 환자들을 보다 쉽게 정리 할 수 있다.

병원 근로자들을 정렬하고 관리 할 수 있다.

## 5. 개발 소스

---

# BoardController.java (PPT 25,26,27,28번 페이지 전부 BoardController)

---

```
package kr.ac.inha.board.board.controller;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.ResponseBody;
import org.springframework.web.servlet.ModelAndView;

import kr.ac.inha.board.board.dto.BoardDto;
import kr.ac.inha.board.board.dto.ReplyDto;
import kr.ac.inha.board.board.service.BoardService;
```

```
@Controller
```

```
public class BoardController {
```

```
@Autowired
```

```
private BoardService boardService;
```

```
@RequestMapping("/board/openBoardList.do")
```

```
public ModelAndView openBoardList() throws Exception {
    ModelAndView mv = new ModelAndView("/board/boardList");
    //List<BoardDto> list = boardService.selectBoardList();
    //mv.addObject("list", list);
    return mv;
}
```

```
@RequestMapping("/board/boardList.ajax")
```

```
@ResponseBody
```

```
public List<BoardDto> boardListAjax() throws Exception {
    List<BoardDto> list = boardService.selectBoardList();
    return list;
}
```

# BoardController.java (PPT 25,26,27,28번 페이지 전부 BoardController)

---

```
@RequestMapping("/board/openBoardWrite.do")
public String openBoardWrite() throws Exception {
    return "/board/boardWrite";
}

@RequestMapping("/board/insertBoard.do")
public String insertBoard(BoardDto board) throws Exception {
    boardService.insertBoard(board);
    return "redirect:/board/openBoardList.do";
}

@RequestMapping("/board/openBoardDetail.do")
public ModelAndView openBoardDetail(@RequestParam int boardIdx) throws Exception {
    ModelAndView mv = new ModelAndView("/board/boardDetail");
    BoardDto board = boardService.selectBoardDetail(boardIdx);
    mv.addObject("board", board);

    List<ReplyDto> replyList = boardService.selectReplyList(boardIdx);
    mv.addObject("replyList", replyList);

    return mv;
}

@RequestMapping("/board/updateBoard.do")
public String updateBoard(BoardDto board) throws Exception {
    boardService.updateBoard(board);
    return "redirect:/board/openBoardList.do";
}

@RequestMapping("/board/deleteBoard.do")
public String deleteBoard(int boardIdx) throws Exception {
    boardService.deleteBoard(boardIdx);
    return "redirect:/board/openBoardList.do";
}

@RequestMapping("/board/bootstrap.do")
public String bootstrap() throws Exception {
    return "/board/bootstrap";
}
```



# BoardController.java (PPT 25,26,27,28번 페이지 전부 BoardController)

---

// 여기서 부터 Reply 컨트롤러

```
@RequestMapping("/board/openReplyList.do")
public ModelAndView openReplyList(int boardIdx) throws Exception {
    ModelAndView mv = new ModelAndView("/board/boardDetail");
    List<ReplyDto> list = boardService.selectReplyList(boardIdx);
    mv.addObject("reply", list);
    return mv;
}
```

// Reply 작성

```
@RequestMapping("/board/openReplyWrite.do")
public ModelAndView openReplyWrite(@RequestParam int boardIdx) throws Exception {
    ModelAndView mv = new ModelAndView("/board/replyWrite");
    mv.addObject("boardIdx", boardIdx);
    return mv;
}
```

// Reply 등록

```
@RequestMapping("/board/insertReply.do")
public String insertReply(ReplyDto board) throws Exception {
    boardService.insertReply(board);
    return "redirect:/board/openBoardDetail.do?boardIdx=" + board.getBoardIdx();
}
```

// Reply 상세창

```
@RequestMapping("/board/openReplyDetail.do")
public ModelAndView openReplyDetail(@RequestParam int replyNo) throws Exception {
    ModelAndView mv = new ModelAndView("/board/replyDetail");
    ReplyDto board = boardService.selectReplyDetail(replyNo);
    mv.addObject("board", board);

    return mv;
}
```

}

# BoardController.java (PPT 25,26,27,28번 페이지 전부 BoardController)

---

// Reply 수정

```
@RequestMapping("/board/updateReply.do")
public String updateReply(ReplyDto reply) throws Exception {
    int boardIdx = getBoardIdxByReplyNo(reply.getReplyNo());
    boardService.updateReply(reply);
    return "redirect:/board/openBoardDetail.do?boardIdx=" + boardIdx;
}
```

```
private int getBoardIdxByReplyNo(int replyNo) throws Exception {
    ReplyDto reply = boardService.selectReplyDetail(replyNo);
    int boardIdx = reply.getBoardIdx();
    return boardIdx;
}
```

// Reply 삭제

```
@RequestMapping("/board/deleteReply.do")
public String deleteReply(int replyNo) throws Exception {
    int boardIdx = getBoardIdxByReplyNo(replyNo);
    boardService.deleteReply(replyNo);
    return "redirect:/board/openBoardDetail.do?boardIdx=" + boardIdx;
}
}
```

# StaffController.java

(PPT 29,30번 페이지 전부 StaffController)

```
package kr.ac.inha.board.board.controller;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.ResponseBody;
import org.springframework.web.servlet.ModelAndView;

import kr.ac.inha.board.board.dto.StaffDto;
import kr.ac.inha.board.board.service.StaffService;

@Controller
public class StaffController {
    @Autowired
    private StaffService staffService;

    //직원 리스트
    @RequestMapping("/board/openStaffList.do")
    public ModelAndView openStaffList() throws Exception {
        ModelAndView mv = new ModelAndView("/board/StaffList");

        return mv;
    }

    @RequestMapping("/board/staffList.ajax")
    @ResponseBody
    public List<StaffDto> staffListAjax() throws Exception {
        List<StaffDto> list = staffService.selectStaffList();
        return list;
    }

    @RequestMapping("/board/openStaffWrite.do")
    public String openStaffWrite() throws Exception {
        return "/board/staffWrite";
    }
}
```

# StaffController.java

(PPT 29,30번 페이지 전부 StaffController)

---

```
@RequestMapping("/board/insertStaff.do")
public String insertStaff(StaffDto board) throws Exception {
    staffService.insertStaff(board);
    return "redirect:/board/openStaffListt.do";
}

@RequestMapping("/board/openStaffDetail.do")
public ModelAndView openStaffDetail(@RequestParam int staffNo) throws Exception {
    ModelAndView mv = new ModelAndView("/board/staffDetail");
    StaffDto board = staffService.selectStaffDetail(staffNo);
    mv.addObject("board", board);

    return mv;
}

@RequestMapping("/board/updateStaff.do")
public String updateStaff(StaffDto staff) throws Exception {
    staffService.updateStaff(staff);
    return "redirect:/board/openStaffListt.do";
}

@RequestMapping("/board/deleteStaff.do")
public String deleteStaff(int staffNo) throws Exception {
    staffService.deleteStaff(staffNo);
    return "redirect:/board/openStaffListt.do";
}

}
```

## BoradDto.java

```
package kr.ac.inha.board.board.dto;

import lombok.Data;

@Data
public class BoardDto {
    private int boardIdx;
    private String title;
    private String disease;
    private String contents;
    private String clinic;
    private String money;
    private String birth;
    private int hitCnt;
    private int replyCnt;
    private String createdDatetime;
    private String creatorId;
    private String doctor;
    private String nurse;
    private String updateDatetime;
    private String updaterId;
    private String deletedYn;
}
```

## ReplyDto.java

```
package kr.ac.inha.board.board.dto;

import lombok.Data;

@Data
public class ReplyDto {
    private int replyNo;
    private int boardIdx;
    private String replyMemo;
    private String createdDatetime;
    private String creatorId;
}
```

## StaffDto.java

```
package kr.ac.inha.board.board.dto;

import lombok.Data;

@Data
public class StaffDto {
    private int staffNo;
    private String name;
    private String birth;
    private String phone;
    private String date;
    private String contents;
    private int hitCnt;
    private String createdDatetime;
    private String creatorId;
    private String updateDatetime;
    private String updaterId;
    private String deletedYn;
}
```

# BoardMapper.java

---

```
package kr.ac.inha.board.board.mapper;

import java.util.List;

import org.apache.ibatis.annotations.Mapper;

import kr.ac.inha.board.board.dto.BoardDto;

@Mapper
public interface BoardMapper {

    List<BoardDto> selectBoardList() throws Exception;

    void insertBoard(BoardDto board) throws Exception;

    BoardDto selectBoardDetail(int boardIdx) throws Exception;

    void updateHitCount(int boardIdx) throws Exception;

    void updateBoard(BoardDto board) throws Exception;

    void deleteBoard(int boardIdx) throws Exception;

    void addReplyCount(int boardIdx) throws Exception;

    void minusReplyCount(int boardIdx) throws Exception;

}
```

# ReplyMapper.java

---

```
package kr.ac.inha.board.board.mapper;

import java.util.List;

import org.apache.ibatis.annotations.Mapper;
import org.apache.ibatis.annotations.Param;

import kr.ac.inha.board.board.dto.ReplyDto;

@Mapper
public interface ReplyMapper {
    List<ReplyDto> selectReplyList(int boardIdx) throws Exception;

    void insertReply(ReplyDto board) throws Exception;

    ReplyDto selectReplyDetail(int replyNo) throws Exception;

    void updateReplyCount(@Param("boardIdx") int boardIdx, @Param("interval") int interval) throws Exception;

    void updateReply(ReplyDto reply) throws Exception;

    void deleteReply(int replyNo) throws Exception;
}
```

# StaffMapper.java

---

```
package kr.ac.inha.board.board.mapper;

import java.util.List;

import org.apache.ibatis.annotations.Mapper;

import kr.ac.inha.board.board.dto.StaffDto;

@Mapper
public interface StaffMapper {

    List<StaffDto> selectStaffList() throws Exception;

    void insertStaff(StaffDto board) throws Exception;

    StaffDto selectStaffDetail(int staffNo) throws Exception;

    void updateHitCount(int staffNo) throws Exception;

    void updateStaff(StaffDto staff) throws Exception;

    void deleteStaff(int staffNo) throws Exception;

    void addReplyCount(int staffNo) throws Exception;

    void minusReplyCount(int staffNo) throws Exception;

}
```



# BoardService.java

---

```
package kr.ac.inha.board.board.service;

import java.util.List;

import kr.ac.inha.board.board.dto.BoardDto;
import kr.ac.inha.board.board.dto.ReplyDto;

public interface BoardService {
    List<BoardDto> selectBoardList() throws Exception;
    void insertBoard(BoardDto board) throws Exception;
    BoardDto selectBoardDetail(int boardIdx ) throws Exception;
    void updateBoard(BoardDto board ) throws Exception;
    void deleteBoard(int boardIdx ) throws Exception;

    List<ReplyDto> selectReplyList(int boardIdx) throws Exception;
    void insertReply(ReplyDto board) throws Exception;
    ReplyDto selectReplyDetail(int replyNo) throws Exception;
    void updateReply(ReplyDto reply) throws Exception;
    void deleteReply(int replyNo ) throws Exception;

}
```

# BoardServiceImpl.java (PPT 36,37,38번 페이지 전부 BoardServiceImpl)

---

```
package kr.ac.inha.board.board.service;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import kr.ac.inha.board.board.dto.BoardDto;
import kr.ac.inha.board.board.dto.ReplyDto;

import kr.ac.inha.board.board.mapper.BoardMapper;
import kr.ac.inha.board.board.mapper.ReplyMapper;

@Service
public class BoardServiceImpl implements BoardService {
    @Autowired
    private BoardMapper boardMapper;
    @Autowired
    private ReplyMapper replyMapper;

    @Override
    public List<BoardDto> selectBoardList() throws Exception {
        List<BoardDto> selectBoardList = boardMapper.selectBoardList();
        for (BoardDto count : selectBoardList) {
            System.out.println(count.getTitle());
        }
        return selectBoardList;
    }

    @Override
    public void insertBoard(BoardDto board) throws Exception {
        boardMapper.insertBoard(board);
    }
}
```

# BoardServiceImpl.java (PPT 36,37,38번 페이지 전부 BoardServiceImpl)

---

```
@Override
public BoardDto selectBoardDetail(int boardIdx) throws Exception {
    BoardDto board = boardMapper.selectBoardDetail(boardIdx);
    boardMapper.updateHitCount(boardIdx);

    return board;
}

@Override
public void updateBoard(BoardDto board) throws Exception {
    boardMapper.updateBoard(board);
}

@Override
public void deleteBoard(int boardIdx) throws Exception {
    boardMapper.deleteBoard(boardIdx);
}

// 여기서부터 Reply serviceImpl
@Override
public List<ReplyDto> selectReplyList(int boardIdx) throws Exception {
    return replyMapper.selectReplyList(boardIdx);
}

@Override
public void insertReply(ReplyDto board) throws Exception {
    replyMapper.insertReply(board);
    boardMapper.addReplyCount(board.getBoardIdx());
}

@Override
public ReplyDto selectReplyDetail(int replyNo) throws Exception {
    return replyMapper.selectReplyDetail(replyNo);
}
```

# BoardServiceImpl.java (PPT 36,37,38번 페이지 전부 BoardServiceImpl)

---

```
@Override
public void updateReply(ReplyDto reply) throws Exception {
    replyMapper.updateReply(reply);
}
```

```
@Override
public void deleteReply(int replyNo) throws Exception {
    ReplyDto board = replyMapper.selectReplyDetail(replyNo);
    replyMapper.deleteReply(replyNo);
    boardMapper.minusReplyCount(board.getBoardIdx());
}
```

```
}
```

# StaffService.java

---

```
package kr.ac.inha.board.board.service;

import java.util.List;

import kr.ac.inha.board.board.dto.StaffDto;

public interface StaffService {

    List<StaffDto> selectStaffList() throws Exception;
    void insertStaff(StaffDto board) throws Exception;
    StaffDto selectStaffDetail(int staffNo ) throws Exception;
    void updateStaff(StaffDto staff ) throws Exception;
    void deleteStaff(int staffNo ) throws Exception;

}
```

# StaffServiceImpl.java

(PPT 40,41번 페이지 전부 StaffServiceImpl)

```
package kr.ac.inha.board.board.service;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import kr.ac.inha.board.board.dto.StaffDto;
import kr.ac.inha.board.board.mapper.StaffMapper;

@Service
public class StaffServiceImpl implements StaffService {

    @Autowired
    private StaffMapper staffMapper;

    //Staff
    @Override
    public List<StaffDto> selectStaffList() throws Exception {
        List<StaffDto> selectStaffList = staffMapper.selectStaffList();
        for (StaffDto count : selectStaffList) {
            System.out.println(count.getName());
        }
        return selectStaffList;
    }

    @Override
    public void insertStaff(StaffDto board) throws Exception {
        staffMapper.insertStaff(board);
    }
}
```

```
@Override
public StaffDto selectStaffDetail(int staffNo) throws Exception {
    StaffDto board = staffMapper.selectStaffDetail(staffNo);
    staffMapper.updateHitCount(staffNo);

    return board;
}

@Override
public void updateStaff(StaffDto staff) throws Exception {
    staffMapper.updateStaff(staff);
}

@Override
public void deleteStaff(int staffNo) throws Exception {
    staffMapper.deleteStaff(staffNo);
}
}
```

# ChartController.java

---

```
package kr.ac.inha.board.chart.controller;

import java.sql.Timestamp;
import java.util.List;

import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.ResponseBody;
import org.springframework.web.servlet.ModelAndView;
import kr.ac.inha.board.chart.dto.StockDto;
import kr.ac.inha.board.chart.service.ChartService;

@Controller
public class ChartController {
    Logger log = LoggerFactory.getLogger(this.getClass());
    @Autowired
    private ChartService chartService;

    @RequestMapping("/chart/chartJs.do")
    public ModelAndView chartJs() throws Exception {
        ModelAndView mv = new ModelAndView("/chart/chartJs");
        mv.addObject("timestamp", new Timestamp(System.currentTimeMillis()));
        return mv;
    }

    @RequestMapping("/chart/amchart.do")
    public ModelAndView amchart() throws Exception {
        ModelAndView mv = new ModelAndView("/chart/amchart");
        mv.addObject("timestamp", new Timestamp(System.currentTimeMillis()));
        return mv;
    }

    @RequestMapping("/chart/stockList.ajax")
    @ResponseBody
    public List<StockDto> stockList(String jongmokCode) throws Exception {
        List<StockDto> list = chartService.selectStockByJongmok(jongmokCode);
        return list;
    }
}
```



# StockDto.java

---

```
package kr.ac.inha.board.chart.dto;
```

```
import lombok.Data;
```

```
@Data
```

```
public class StockDto {  
    private String jongmokCode;  
    private String date;  
    private int open;  
    private int high;  
    private int low;  
    private int close;  
    private long volume;  
}
```

# ChartMapper.java

---

```
package kr.ac.inha.board.chart.mapper;

import java.util.List;

import org.apache.ibatis.annotations.Mapper;

import kr.ac.inha.board.chart.dto.StockDto;

@Mapper
public interface ChartMapper {
    List<StockDto> selectStockByJongmok(String jongmokCode) throws Exception;
}
```

## ChartService.java

---

```
package kr.ac.inha.board.chart.service;

import java.util.List;

import kr.ac.inha.board.chart.dto.StockDto;

public interface ChartService {
    List<StockDto> selectStockByJongmok(String jongmokCode) throws Exception;
}
```

## ChartServiceImpl.java

---

```
package kr.ac.inha.board.chart.service;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import kr.ac.inha.board.chart.dto.StockDto;
import kr.ac.inha.board.chart.mapper.ChartMapper;

@Service
public class ChartServiceImpl implements ChartService{
    @Autowired
    private ChartMapper chartMapper;

    @Override
    public List<StockDto> selectStockByJongmok(String jongmokCode) throws Exception {
        return chartMapper.selectStockByJongmok(jongmokCode);
    }
}
```

# LoginController.java

(PPT 46,47,48번 페이지 전부 LoginController)

```
package kr.ac.inha.board.login.controller;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.ResponseBody;
import org.springframework.web.servlet.ModelAndView;
import com.fasterxml.jackson.databind.ObjectMapper;
import kr.ac.inha.board.login.dto.MembersDto;
import kr.ac.inha.board.login.dto.RegistDto;
import kr.ac.inha.board.login.dto.ResultDto;
import kr.ac.inha.board.login.service.LoginService;
import lombok.extern.slf4j.Slf4j;
```

```
@Slf4j
```

```
@Controller
```

```
public class LoginController {
```

```
@Autowired
```

```
private LoginService loginService;
```

```
@RequestMapping("/login/login.do")
```

```
public ModelAndView login() throws Exception {
```

```
ModelAndView mv = new ModelAndView("board/login/login");
```

```
return mv;
```

```
}
```

```
@RequestMapping("/login/register.do")
```

```
public ModelAndView register() throws Exception {
```

```
ModelAndView mv = new ModelAndView("board/login/register");
```

```
return mv;
```

```
}
```

# LoginController.java

(PPT 46,47,48번 페이지 전부 LoginController)

---

```
@RequestMapping(value="/login/register.ajax", method=RequestMethod.POST, produces = "application/json; charset=utf8")
@ResponseBody
public String saveRegisterInfo(@RequestBody RegistDto mbr) throws Exception{
    ObjectMapper mapper = new ObjectMapper();

    ResultDto result = new ResultDto();

    Log.info(mbr.toString());

    String rtn = loginService.saveRegisterInfo(mbr);
    if (rtn.equals("Ok")) {
        result.setStatus("Ok");
        result.setErrMsg("");
    }
    else {
        result.setStatus("Error");
        result.setErrMsg(rtn);
    }

    String jsonString = mapper.writeValueAsString(result);

    Log.info("result = " + result);

    return jsonString;
}
```

```
@RequestMapping(value="/login/certification.do", method=RequestMethod.GET)
@ResponseBody
public ModelAndView certification(MembersDto mbr) throws Exception{
    ModelAndView mv = null;

    Log.info(mbr.toString());

    String rtn = loginService.saveCertification(mbr);
    if (rtn.equals("Ok")) {
        mv = new ModelAndView("board/login/welcome");
    }
    else {
        mv = new ModelAndView("board/login/denied");
        mv.addObject("errMsg", rtn);
    }

    Log.info("result = " + rtn);

    return mv;
}
}
```

## MembersDto.java

```
package kr.ac.inha.board.login.dto;

import lombok.Data;

@Data
public class MembersDto {
    private String userNo;
    private String userId;
    private String userPass;
    private String firstName;
    private String lastName;
    private String roleName;
    private String enabled;
    private String registDate;
    private String certificationDate;
}
```

## RegistDto.java

```
package kr.ac.inha.board.login.dto;

import lombok.Data;

@Data
public class RegistDto {
    private String firstName;
    private String lastName;
    private String inputEmail;
    private String inputPassword;
    private String confirmPassword;
}
```

## ResultDto.java

```
package kr.ac.inha.board.login.dto;

import lombok.Data;

@Data
public class ResultDto {
    private String status;
    private String errMsg;
}
```

# LoginMapper.java

---

```
package kr.ac.inha.board.login.mapper;

import java.util.List;

import org.apache.ibatis.annotations.Mapper;

import kr.ac.inha.board.login.dto.MembersDto;

@Mapper
public interface LoginMapper {
    List<MembersDto> selectMembers(String userId) throws Exception;
    void insertMembers(MembersDto membersDto) throws Exception;
    void updateMembersCert(MembersDto membersDto) throws Exception;
}
```

# LoginService.java

---

```
package kr.ac.inha.board.login.service;

import kr.ac.inha.board.login.dto.MembersDto;
import kr.ac.inha.board.login.dto.RegistDto;

public interface LoginService {
    String saveRegisterInfo(RegistDto mbr) throws Exception;
    String saveCertification(MembersDto mbr) throws Exception;
}
```



# LoginServiceImpl.java (PPT 51,52,53,54,55번 페이지 전부 LoginServiceImpl)

---

```
package kr.ac.inha.board.login.service;

import java.util.List;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.mail.javamail.JavaMailSender;
import org.springframework.stereotype.Service;

import kr.ac.inha.board.login.dto.MembersDto;
import kr.ac.inha.board.login.dto.RegistDto;
import kr.ac.inha.board.login.mapper.LoginMapper;

@Service
public class LoginServiceImpl implements LoginService {
    @Autowired
    private LoginMapper loginMapper;

    @Autowired
    private JavaMailSender emailSender;

    @Override
    public String saveRegisterInfo(RegistDto mbr) throws Exception {
        String emailRegExp =
            "^[_A-Za-z0-9-\\+](\\.[_A-Za-z0-9-]+)*@" +
            "[A-Za-z0-9-]+(\\.[A-Za-z0-9]+)*(\\.[A-Za-z]{2,})$";
        Pattern pattern;

        pattern = Pattern.compile(emailRegExp);
```

# LoginServiceImpl.java (PPT 51,52,53,54,55번 페이지 전부 LoginServiceImpl)

---

```
if (mbr.getFirstName() == null || mbr.getFirstName().trim().isEmpty()) {
    return "이름(First Name)은 필수 정보 입니다.";
}
if (mbr.getLastName() == null || mbr.getLastName().trim().isEmpty()) {
    return "성(Last Name)은 필수 정보 입니다.";
}
if (mbr.getInputEmail() == null || mbr.getInputEmail().trim().isEmpty()) {
    return "이메일(Email address)은 필수 정보 입니다.";
}
else {
    Matcher matcher = pattern.matcher(mbr.getInputEmail());
    if (!matcher.matches()) {
        return "이메일 형식이 올바르지 않습니다.";
    }
}
if (mbr.getInputPassword() == null || mbr.getInputPassword().trim().isEmpty()) {
    return "비밀번호>Password)은 필수 정보 입니다.";
}
if (mbr.getInputPassword().trim().length() < 8) {
    return "비밀번호>Password)는 8자 이상 입력하세요.";
}
if (mbr.getConfirmPassword() == null || mbr.getConfirmPassword().trim().isEmpty()) {
    return "비밀번호확인(Confirm password)은 필수 정보 입니다.";
}
if (!mbr.getInputPassword().equals(mbr.getConfirmPassword())) {
    return "비밀번호가 일치하지 않습니다.";
}
```

# LoginServiceImpl.java (PPT 51,52,53,54,55번 페이지 전부 LoginServiceImpl)

---

// 등록상태 확인

```
List<MembersDto> membersList = loginMapper.selectMembers(mbr.getInputEmail());
if (!membersList.isEmpty()) {
    MembersDto members = membersList.get(0);
    if (members.getCertificationDate() == null || members.getCertificationDate().equals("")) {
        return "이미 신청중인 상태이나 이메일 인증이 완료되지 않았습니다.\n발송된 이메일을 열어서 인증 처리하시기 바랍니다.";
    }
    else if (!members.getEnabled().equals("Y")) {
        return "계정이 비활성(사용불가) 상태입니다.";
    }
    else {
        return "이미 신청이 완료되었습니다.\n로그인 하셔서 사용하시기 바랍니다.";
    }
}
```

// 계정 등록

```
MembersDto members = new MembersDto();
members.setUserId(mbr.getInputEmail());
members.setFirstName(mbr.getFirstName());
members.setLastName(mbr.getLastName());
members.setUserPass(mbr.getInputPassword());
```

```
loginMapper.insertMembers(members);
```

```
membersList = loginMapper.selectMembers(mbr.getInputEmail());
members = membersList.get(0);
```

# LoginServiceImpl.java (PPT 51,52,53,54,55번 페이지 전부 LoginServiceImpl)

---

```
// 이메일 발송
```

```
try {  
String htmlStr = "<h2>" + members.getLastName() + members.getFirstName() + "님 안녕하세요.</h2>"  
+ "<h3><p>사이트 로그인을 하시려면, 인증하기 버튼을 눌러서 인증 처리를 하시기 바랍니다. -&gt; "  
+ "<a href='http://localhost:8080/login/certification.do?userNo=" + members.getUserNo() + "&userId=" + members.getUserId()  
+ "&userKey=18f198e0-92a9-447c-9adf-b93b4b578644'>[인증하기]</a></p>"  
+ "(혹시 잘못 전달된 메일이라면 이 이메일을 무시하셔도 됩니다)</h3>";  

```

```
MailHandler mail = new MailHandler(emailSender);  
mail.setFrom("qkrgml dbs123@gmail.com", "xx병원");  
mail.setTo(mbr.getInputEmail());  
mail.setSubject("[본인인증] 회원가입 인증요청");  
mail.setText(htmlStr);  
mail.send();  
} catch (Exception e) {  
e.printStackTrace();  
throw e;  
}  

```

```
return "Ok";  
}
```

```
@Override
```

```
public String saveCertification(MembersDto mbr) throws Exception {  
// 등록상태 확인  
List<MembersDto> membersList = loginMapper.selectMembers(mbr.getUserId());  
if (membersList.isEmpty()) {  
return "신청한 내역이 존재하지 않습니다.";  
}  
}
```

# LoginServiceImpl.java (PPT 51,52,53,54,55번 페이지 전부 LoginServiceImpl)

---

```
MembersDto members = membersList.get(0);
if (!members.getUserNo().equals(mbr.getUserNo())) {
    return "신청한 계정 정보와 인증 정보가 불일치합니다.";
}
else if (members.getCertificationDate() != null && !members.getCertificationDate().equals("")) {
    if (!members.getEnabled().equals("Y")) {
        return "계정이 비활성(사용불가) 상태입니다.";
    }
    return "Ok";
}
else if (members.getEnabled().equals("Y") && members.getCertificationDate() != null) {
    return "Ok";
}

// 계정 인증
loginMapper.updateMembersCert(members);

return "Ok";
}
}
```

# MailHandler.java

(PPT 56,57번 페이지 전부 MailHandler)

```
package kr.ac.inha.board.login.service;

import java.io.File;
import java.io.UnsupportedEncodingException;

import javax.mail.MessagingException;
import javax.mail.internet.MimeMessage;
import javax.activation.DataSource;

import org.springframework.core.io.Resource;
import org.springframework.mail.javamail.JavaMailSender;
import org.springframework.mail.javamail.MimeMessageHelper;

public class MailHandler {
    private JavaMailSender sender;
    private MimeMessage message;
    private MimeMessageHelper messageHelper;

    // MailHandler의 생성자
    public MailHandler(JavaMailSender jSender) throws MessagingException {
        this.sender = jSender;
        message = jSender.createMimeMessage();
        messageHelper = new MimeMessageHelper(message, true, "UTF-8");
        // MimeMessageHelper의 생성자 두번째 파라미터는 다수의 사람에게 보낼 수 있는 설정, 세번째는 기본 인코딩 방식
    }

    // 이 이메일이 누구로부터 가는가..
    public void setFrom(String email, String name) throws UnsupportedEncodingException, MessagingException {
        messageHelper.setFrom(email, name);
    }

    // 누구에게 보낼 것인가..
    public void setTo(String email) throws MessagingException {
        messageHelper.setTo(email);
    }
}
```

# MailHandler.java

(PPT 56,57번 페이지 전부 MailHandler)

```
// 보낼때 제목
public void setSubject(String subject) throws MessagingException {
    messageHelper.setSubject(subject);
}

// 보낼 메일의 내용.. 두번째 파라미터는 html을 적용할 것인가 아닌가. true시 html형식으로 작성하면 html형식으로 보임..
public void setText(String text) throws MessagingException {
    messageHelper.setText(text, true);
}

//addInline메서드는 간단한 첨부파일을 보내는데 사용..
public void addInline(String contentId, Resource resource) throws MessagingException {
    messageHelper.addInline(contentId, resource);
}

public void addInline(String contentId, File file) throws MessagingException {
    messageHelper.addInline(contentId, file);
}

public void addInline(String contentId, DataSource source) throws MessagingException {
    messageHelper.addInline(contentId, source);
}

// 실제로 메일을 보내는 메서드..
public void send() {
    try {
        sender.send(message);
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

# SecurityConfig.java

(PPT 58,59번 페이지 전부 SecurityConfig)

---

```
package kr.ac.inha.board.security;

import javax.sql.DataSource;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.context.annotation.Bean;
import org.springframework.security.config.annotation.authentication.builders.AuthenticationManagerBuilder;
import org.springframework.security.config.annotation.web.builders.HttpSecurity;
import org.springframework.security.config.annotation.web.configuration.EnableWebSecurity;
import org.springframework.security.config.annotation.web.configuration.WebSecurityConfigurerAdapter;
import org.thymeleaf.extras.springsecurity5.dialect.SpringSecurityDialect;

import lombok.extern.slf4j.Slf4j;

@Slf4j
@EnableWebSecurity
public class SecurityConfig extends WebSecurityConfigurerAdapter {
    @Autowired
    DataSource dataSource;

    @Override
    protected void configure(HttpSecurity http) throws Exception {
        log.info("security config....");
        // @RequestMapping 별 권한제어

        http.authorizeRequests().antMatchers("/board/login/**").permitAll();
        http.authorizeRequests().antMatchers("/board/**").hasAnyRole("ADMIN", "USER", "GUEST");

        http.authorizeRequests().antMatchers("/login/register.ajax").permitAll();
        http.csrf().ignoringAntMatchers("/login/register.ajax").ignoringRequestMatchers();
```



```
// 사용자 정의 로그인 페이지
http.formLogin().loginPage("/login/login.do");

// 권한 없을때 띄울 페이지
// http.exceptionHandling().accessDeniedPage("/board/openBoardList3.do");

// 로그아웃시 지정 페이지로 이동
http.logout().logoutSuccessUrl("/board/openBoardListt.do").invalidateHttpSession(true);
}

@Autowired
public void configureGlobal(AuthenticationManagerBuilder auth) throws Exception {
    //auth.inMemoryAuthentication().withUser("admin@inha.ac.kr").password("{noop}admin!123").roles("ADMIN");
    //auth.inMemoryAuthentication().withUser("user@inha.ac.kr").password("{noop}user!123").roles("USER");
    //auth.inMemoryAuthentication().withUser("guest@inha.ac.kr").password("{noop}guest!123").roles("GUEST");

    String query1 = "SELECT user_id username, CONCAT('{noop}', user_pass) password, "
    + " CASE WHEN enabled = 'Y' THEN true else false END enabled" + " FROM members WHERE user_id = ?";
    String query2 = "SELECT user_id, role_name role" + " FROM members WHERE user_id = ?";

    auth.jdbcAuthentication().dataSource(dataSource).usersByUsernameQuery(query1).rolePrefix("ROLE_")
    .authoritiesByUsernameQuery(query2);
}

@Bean
public SpringSecurityDialect springSecurityDialect() {
    return new SpringSecurityDialect();
}
}
```

# sql-board.xml

---

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE mapper PUBLIC "-//mybatis.org/DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-mapper.dtd">

<mapper namespace="kr.ac.inha.board.board.mapper.BoardMapper">
  <select id="selectBoardList"
    resultType="kr.ac.inha.board.board.dto.BoardDto">
    <![CDATA[
      SELECT
        board_idx,
        title,
        disease,
        birth,
        doctor,
        nurse,
        money,

        hit_cnt,
        reply_cnt,
        DATE_FORMAT(created_datetime, '%Y-%m-%d %H:%i:%s') AS created_datetime
      FROM
        t_board
      WHERE
        deleted_yn = 'N'
      ORDER BY board_idx DESC
    ]]>
  </select>

  <insert id="insertBoard" parameterType="kr.ac.inha.board.board.dto.BoardDto" useGeneratedKeys="true" keyProperty="boardIdx">
    <![CDATA[
      INSERT INTO t_board
      (
        title,
        disease,
        birth,
        doctor,
        nurse,
        money,
        contents,
        clinic,
        created_datetime,
        creator_id
      )
      VALUES
      (
        #{title},
        #{disease},
        #{birth},
        #{doctor},
        #{nurse},
        #{money},
        #{contents},
        #{clinic},
        NOW(),
        'admin'
      )
    ]]>
  </insert>
```

# sql-board.xml

---

```
<select id="selectBoardDetail" parameterType="int"
resultType="kr.ac.inha.board.board.dto.BoardDto">
<![CDATA[
SELECT
board_idx,
title,
disease,
birth,
contents,
clinic,
doctor,
nurse,
money,
hit_cnt,
reply_cnt,
DATE_FORMAT(created_datetime, '%Y.%m.%d %H:%i:%s') AS created_datetime,
creator_id
FROM
t_board
WHERE
board_idx = #{boardIdx}
AND deleted_yn = 'N'
]]>
</select>
```

```
<update id="updateHitCount" parameterType="int">
<![CDATA[
UPDATE
t_board
SET
hit_cnt = hit_cnt + 1
WHERE
board_idx = #{boardIdx}
]]>
</update>
```

```
<update id="updateBoard"
parameterType="kr.ac.inha.board.board.dto.BoardDto">
<![CDATA[
UPDATE t_board
SET title = #{title},
disease = #{disease},
birth = #{birth},
doctor = #{doctor},
nurse = #{nurse},
money = #{money},
contents = #{contents},
clinic = #{clinic},
updated_datetime = Now(),
updater_id = 'admin'
WHERE
board_idx = #{boardIdx}
]]>
</update>
```

# sql-board.xml

---

```
<delete id="deleteBoard" parameterType="int">
<![CDATA[
UPDATE t_board
SET deleted_yn = 'Y',
updated_datetime = NOW(),
updater_id= 'admin'
WHERE
board_idx = #{boardIdx}
]]>
</delete>
```

```
<update id="addReplyCount" parameterType="int">
```

```
<![CDATA[
UPDATE
t_board
SET
reply_cnt = reply_cnt + 1
WHERE
board_idx = #{boardIdx}
]]>
</update>
```

```
<update id="minusReplyCount" parameterType="int">
```

```
<![CDATA[
UPDATE
t_board
SET
reply_cnt = reply_cnt - 1
WHERE
board_idx = #{boardIdx}
]]>
</update>
```

```
</mapper>
```

# sql-login.xml

---

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-mapper.dtd">

<mapper namespace="kr.ac.inha.board.Login.mapper.LoginMapper">
<select id="selectMembers" parameterType="String" resultType="kr.ac.inha.board.Login.dto.MembersDto">
SELECT user_no,
       user_id,
       user_pass,
       first_name,
       last_name,
       role_name,
       enabled,
       DATE_FORMAT(regist_date, '%Y-%m-%d') regist_date,
       DATE_FORMAT(certification_date, '%Y-%m-%d') certification_date
FROM members
WHERE user_id = #{userId}
</select>

<insert id="insertMembers" parameterType="kr.ac.inha.board.Login.dto.MembersDto">
INSERT INTO members
  (user_id, user_pass, first_name, last_name, role_name, enabled, regist_date, certification_date)
VALUES (#{userId},
       #{userPass},
       #{firstName},
       #{lastName},
       'USER',
       'N',
       NOW(),
       null)
</insert>

<update id="updateMembersCert" parameterType="kr.ac.inha.board.Login.dto.MembersDto">
UPDATE members
  SET enabled = 'Y',
      certification_date = NOW()
WHERE user_id = #{userId}
</update>
</mapper>
```

# sql-reply.xml

---

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-mapper.dtd">

<mapper namespace="kr.ac.inha.board.board.mapper.ReplyMapper">

<select id="selectReplyList"
resultType="kr.ac.inha.board.board.dto.ReplyDto">
<![CDATA[
SELECT
reply_no,
board_idx,
reply_memo,
DATE_FORMAT(created_datetime, '%Y-%m-%d %H:%i:%s') AS created_datetime,
creator_id
FROM
t_reply
WHERE
board_idx = #{boardIdx}
ORDER BY created_datetime DESC
]]>
</select>

<select id="selectReplyDetail"
resultType="kr.ac.inha.board.board.dto.ReplyDto">
<![CDATA[
SELECT
reply_no,
board_idx,
reply_memo,
DATE_FORMAT(created_datetime, '%Y-%m-%d %H:%i:%s') AS created_datetime,
creator_id
FROM
t_reply
WHERE
reply_no = #{replyNo}
]]>
</select>
```

# sql-reply.xml

---

```
<insert id="insertReply"
parameterType="kr.ac.inha.board.board.dto.ReplyDto"
useGeneratedKeys="true" keyProperty="replyNo">
<![CDATA[
INSERT INTO t_reply
(
board_idx,
reply_memo,
created_datetime,
creator_id
)
VALUES
(
#{boardIdx},
#{replyMemo},
NOW(),
'admin'
)
]]>
</insert>

<update id="updateReplyCount">
<![CDATA[
UPDATE t_board
SET reply_cnt = ifnull(reply_cnt, 0) + #{interval}
WHERE board_idx = #{boardIdx}
]]>
</update>

<update id="updateReply"
parameterType="kr.ac.inha.board.board.dto.ReplyDto">
<![CDATA[
UPDATE t_reply
SET reply_memo = #{replyMemo}
WHERE
reply_no = #{replyNo}
]]>
</update>

<delete id="deleteReply" parameterType="int">
<![CDATA[
DELETE FROM t_reply
WHERE reply_no = #{replyNo}
]]>
</delete>
</mapper>
```

# sql-staff.xml

---

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE mapper PUBLIC "-//mybatis.org/DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-mapper.dtd">

<mapper namespace="kr.ac.inha.board.board.mapper.StaffMapper">
  <select id="selectStaffList"
    resultType="kr.ac.inha.board.board.dto.StaffDto">
    <![CDATA[
      SELECT
        staff_no,
        name,
        birth,
        phone,
        date,
        hit_cnt,
        DATE_FORMAT(created_datetime, '%Y-%m-%d %H:%i:%s') AS created_datetime
      FROM
        t_staff
      WHERE
        deleted_yn = 'N'
      ORDER BY staff_no DESC
    ]]>
  </select>

  <insert id="insertStaff" parameterType="kr.ac.inha.board.board.dto.StaffDto" useGeneratedKeys="true" keyProperty="staffNo">
    <![CDATA[
      INSERT INTO t_staff
      (
        name,
        birth,
        phone,
        date,
        contents,
        created_datetime,
        creator_id
      )
      VALUES
      (
        #{name},
        #{birth},
        #{phone},
        #{date},
        #{contents},
        NOW(),
        'admin'
      )
    ]]>
  </insert>
```



# sql-staff.xml

---

```
<select id="selectStaffDetail" parameterType="int"
resultType="kr.ac.inha.board.board.dto.StaffDto">
<![CDATA[
SELECT
staff_no,
name,
birth,
contents,
phone,
date,
hit_cnt,
DATE_FORMAT(created_datetime, '%Y.%m.%d %H:%i:%s') AS created_datetime,
creator_id
FROM
t_staff
WHERE
staff_no = #{staffNo}
AND deleted_yn = 'N'
]]>
</select>

<update id="updateHitCount" parameterType="int">
<![CDATA[
UPDATE
t_staff
SET
hit_cnt = hit_cnt + 1
WHERE
staff_no = #{staffNo}
]]>
</update>

<update id="updateStaff"
parameterType="kr.ac.inha.board.board.dto.StaffDto">
<![CDATA[
UPDATE t_staff
SET name = #{name},
birth = #{birth},
phone = #{phone},
date = #{date},
contents = #{contents},
updated_datetime = Now(),
updater_id = 'admin'
WHERE
staff_no = #{staffNo}
]]>
</update>

<delete id="deleteStaff" parameterType="int">
<![CDATA[
UPDATE t_staff
SET deleted_yn = 'Y',
updated_datetime = NOW(),
updater_id = 'admin'
WHERE
staff_no = #{staffNo}
]]>
</delete>

</mapper>
```

# sql-stock.xml

---

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-mapper.dtd">
<mapper namespace="kr.ac.inha.board.chart.mapper.ChartMapper">
<select id="selectStockByJongmok"
resultType="kr.ac.inha.board.chart.dto.StockDto">
<![CDATA[
SELECT jongmok_code,
DATE_FORMAT(date, '%Y-%m-%d') AS date,
`open`,
high,
low,
`close`,
volume
FROM stock
WHERE jongmok_code = #{jongmokCode}
AND date >= '2020-01-01'
ORDER BY `date`
]]>
</select>
</mapper>
```

# default\_layout.xml

```
<!DOCTYPE html>
<html lang="en"
xmlns:th="http://www.thymeleaf.org"
xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
xmlns:sec="http://www.thymeleaf.org/extras/spring-security">

    <head>
        <meta charset="utf-8" />
        <meta http-equiv="X-UA-Compatible" content="IE=edge" />
        <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
        <meta name="description" content="" />
        <meta name="author" content="" />
        <meta name="_csrf" th:content="{_csrf.token}"/>
        <meta name="_csrf_header" th:content="{_csrf.headerName}"/>
        <title layout:title-pattern="$LAYOUT_TITLE - $CONTENT_TITLE">xx병원</title>
        <link href="/css/styles.css" rel="stylesheet"/>
        <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">
        <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.6.3/css/all.css">

    </head>
    <body>
        <nav class="sb-topnav navbar navbar-expand navbar-dark bg-dark">
            <!-- Navbar Brand-->
            <a class="navbar-brand ps-3" href="/board/openBoardList.do">Patient care</a>
            <!-- Sidebar Toggle-->
            <button class="btn btn-link btn-sm order-1 order-lg-0 me-4 me-lg-0" id="sidebarToggle" href="#"><i class="fas fa-bars"></i></button>
            <!-- Navbar Search-->
            <form class="d-none d-md-inline-block form-inline ms-auto me-0 me-md-3 my-2 my-md-0">
                <div class="input-group">
                    <input class="form-control" type="text" placeholder="Search for..." aria-label="Search for..." aria-describedby="btnNavbarSearch" />
                    <button class="btn btn-primary" id="btnNavbarSearch" type="button"><i class="fas fa-search"></i></button>
                </div>
            </form>
            <!-- Navbar-->
            <!--ul class="navbar-nav ms-auto ms-md-0 me-3 me-lg-4">
                <li class="nav-item dropdown">
                    <a class="nav-link dropdown-toggle" id="navbarDropdown" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false"><i class="fas fa-user fa-fw"></i></a>
                    <ul class="dropdown-menu dropdown-menu-end" aria-labelledby="navbarDropdown">
                        <li><a class="dropdown-item" href="#">Settings</a></li>
                        <li><a class="dropdown-item" href="#">Activity Log</a></li>
                        <li><hr class="dropdown-divider" /></li>
                        <li><a class="dropdown-item" href="#">Logout</a></li>
                    </ul>
                </li>
            </ul-->
            <button sec:authorize="isAnonymous()" type="button" id="goLogin"
class="btn btn-success btn-sm">로그인</button>
            <button sec:authorize="not isAnonymous()" type="button" id="goLogout"
class="btn btn-success btn-sm">로그아웃</button>
        </nav>
```

# default\_layout.xml

---

```
<div id="layoutSidenav">
  <div id="layoutSidenav_nav">
    <nav class="sb-sidenav accordion sb-sidenav-dark" id="sidenavAccordion">
      <div class="sb-sidenav-menu">
        <div class="nav">
          <!--div class="sb-sidenav-menu-heading">Core</div>
          <a class="nav-link" href="index.html">
            <div class="sb-nav-link-icon"><i class="fas fa-tachometer-alt"></i></div>
            Dashboard
          </a>
          <div class="sb-sidenav-menu-heading">Interface</div>
          <a class="nav-link collapsed" href="#" data-bs-toggle="collapse" data-bs-target="#collapseLayouts" aria-expanded="false" aria-controls="collapseLayouts">
            <div class="sb-nav-link-icon"><i class="fas fa-columns"></i></div>
            Layouts
            <div class="sb-sidenav-collapse-arrow"><i class="fas fa-angle-down"></i></div>
          </a>
          <div class="collapse" id="collapseLayouts" aria-labelledby="headingOne" data-bs-parent="#sidenavAccordion">
            <nav class="sb-sidenav-menu-nested nav">
              <a class="nav-link" href="layout-static.html">Static Navigation</a>
              <a class="nav-link" href="layout-sidenav-light.html">Light Sidenav</a>
            </nav>
          </div>
          <a class="nav-link collapsed" href="#" data-bs-toggle="collapse" data-bs-target="#collapsePages" aria-expanded="false" aria-controls="collapsePages">
            <div class="sb-nav-link-icon"><i class="fas fa-book-open"></i></div>
            Pages
            <div class="sb-sidenav-collapse-arrow"><i class="fas fa-angle-down"></i></div>
          </a-->
          <div class="collapse" id="collapsePages" aria-labelledby="headingTwo" data-bs-parent="#sidenavAccordion">
            <nav class="sb-sidenav-menu-nested nav accordion" id="sidenavAccordionPages">
              <a class="nav-link collapsed" href="#" data-bs-toggle="collapse" data-bs-target="#pagesCollapseAuth" aria-expanded="false" aria-controls="pagesCollapseAuth">
                Authentication
                <div class="sb-sidenav-collapse-arrow"><i class="fas fa-angle-down"></i></div>
              </a>
              <div class="collapse" id="pagesCollapseAuth" aria-labelledby="headingOne" data-bs-parent="#sidenavAccordionPages">
                <nav class="sb-sidenav-menu-nested nav">
                  <a class="nav-link" href="login.html">Login</a>
                  <a class="nav-link" href="register.html">Register</a>
                  <a class="nav-link" href="password.html">Forgot Password</a>
                </nav>
              </div>
              <a class="nav-link collapsed" href="#" data-bs-toggle="collapse" data-bs-target="#pagesCollapseError" aria-expanded="false" aria-controls="pagesCollapseError">
                Error
                <div class="sb-sidenav-collapse-arrow"><i class="fas fa-angle-down"></i></div>
              </a>
              <div class="collapse" id="pagesCollapseError" aria-labelledby="headingOne" data-bs-parent="#sidenavAccordionPages">
                <nav class="sb-sidenav-menu-nested nav">
                  <a class="nav-link" href="401.html">401 Page</a>
                  <a class="nav-link" href="404.html">404 Page</a>
                  <a class="nav-link" href="500.html">500 Page</a>
                </nav>
              </div>
            </nav>
          </div>
          <div class="sb-sidenav-menu-heading">Addons</div>
        </div>
      </div>
    </nav>
  </div>
  <div class="sb-sidenav-menu-heading">Addons</div>
</div>
```

# default\_layout.xml

```
<a sec:authorize="hasAnyRole('USER', 'ADMIN')"
```

# default\_layout.xml

---

```
<div class="modal fade" id="logoutModal" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel"
aria-hidden="true">
<div class="modal-dialog" role="document">
<div class="modal-content">
<div class="modal-header">
<h5 class="modal-title" id="exampleModalLabel">Ready to Leave?</h5>
<button class="close" type="button" data-dismiss="modal" aria-label="Close">
<span aria-hidden="true">x</span>
</button>
</div>
<div class="modal-body">Select "Logout" below if you are ready
to end your current session.</div>
<div class="modal-footer">
<button class="btn btn-secondary" type="button" data-dismiss="modal">Cancel</button>
<form method="post" action="/logout">
<input type="hidden" th:name="{_csrf.parameterName}" th:value="{_csrf.token}" />
<button type="submit" class="btn btn-primary">Logout</button>
</form>
</div>
</div>
</div>
</div>
```

```
<script src="https://cdn.jsdelivr.net/npm/jquery/dist/jquery.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"></script>
<script src="/js/scripts.js"></script>
<script>
var token = $("meta[name='_csrf']").attr("content");
var header = $("meta[name='_csrf_header']").attr("content");
$(document).ajaxSend(function(e, xhr, opt){
xhr.setRequestHeader(header, token);
});
$("#goLogin").on('click', function() {
window.location.href = "/login/login.do";
});
$("#goLogout").on('click', function() {
$("#logoutModal").modal();
});
</script>
<!-- custom javascript -->
<th:block layout:fragment="script"></th:block>
</body>
</html>
```

# denied.xml

---

```
<!DOCTYPE html>
<html lang="en"
xmlns:th="http://www.thymeleaf.org"
xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
xmlns:sec="http://www.thymeleaf.org/extras/spring-security"
layout:decorator="board/layout/default_layout">

<head>
<title>xx병원</title>
</head>

<body id="page-top">
<div class="container-fluid" layout:fragment="content">
<br>
<br>
<h3 style="text-align:center">인증이 실패하였습니다.</h3>
<br>
<h5 style="text-align:center;color:red" th:text="${errMsg}">...</h5>
<br>
</div>
<!-- /.container-fluid -->

<th:block layout:fragment="script">
</th:block>
</body>

</html>
```

# login.xml

```
<!DOCTYPE html>
<html lang="en" xmlns:th="http://www.thymeleaf.org">
  <head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
    <meta name="description" content="" />
    <meta name="author" content="" />
    <title>환자 관리 로그인 창</title>
    <link href="/css/styles.css" rel="stylesheet" />
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" crossorigin="anonymous"></script>
  </head>
  <body class="bg-primary">
    <div id="layoutAuthentication">
      <div id="layoutAuthentication_content">
        <main>
          <div class="container">
            <div class="row justify-content-center">
              <div class="col-lg-5">
                <div class="card shadow-lg border-0 rounded-lg mt-5">
                  <div class="card-header"><h3 class="text-center font-weight-light my-4">Patient care Login</h3></div>
                  <div class="card-body">
                    <form method="post">
                      <div class="form-floating mb-3">
                        <input class="form-control" id="username" name="username" type="email" placeholder="name@example.com" />
                        <label for="inputEmail">이메일을 입력해주세요.</label>
                      </div>
                      <div class="form-floating mb-3">
                        <input class="form-control" id="password" name="password" type="password" placeholder="Password" />
                        <label for="inputPassword">비밀번호를 입력해주세요.</label>
                      </div>
                      <!--div class="form-check mb-3">
                        <input class="form-check-input" id="inputRememberPassword" type="checkbox" value="" />
                        <label class="form-check-label" for="inputRememberPassword">Remember Password</label>
                      </div>
                      <div class="d-flex align-items-center justify-content-between mt-4 mb-0">
                        <a class="small" href="password.html">Forgot Password</a>

                        </div-->
                      <input type="hidden" th:name="${_csrf.parameterName}" th:value="${_csrf.token}"/>
                      <button type="submit" class="btn btn-primary btn-block">로그인</button>
                    </form>
                  </div>
                  <div class="card-footer text-center py-3">
                    <div class="small"><a href="/login/register.do">처음 시작이라면 회원가입을 해주세요!</a></div>
                  </div>
                </div>
              </div>
            </div>
          </div>
        </main>
      </div>
      <div id="layoutAuthentication_footer">
        <footer class="py-4 bg-light mt-auto">
          <div class="container-fluid px-4">
            <div class="d-flex align-items-center justify-content-between small">
              <div class="text-muted">phy &copy; Your Website 2021</div>
            </div>
          </div>
        </footer>
      </div>
    </div>
  </body>
</html>
```



# password.xml

---

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
    <meta name="description" content="" />
    <meta name="author" content="" />
    <title>Password Reset - SB Admin</title>
    <link href="css/styles.css" rel="stylesheet" />
    <script src="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/js/all.min.js" crossorigin="anonymous"></script>
  </head>
  <body class="bg-primary">
    <div id="layoutAuthentication">
      <div id="layoutAuthentication_content">
        <main>
          <div class="container">
            <div class="row justify-content-center">
              <div class="col-lg-5">
                <div class="card shadow-lg border-0 rounded-lg mt-5">
                  <div class="card-header"><h3 class="text-center font-weight-light my-4">Password Recovery</h3></div>
                  <div class="card-body">
                    <div class="small mb-3 text-muted">Enter your email address and we will send you a link to reset your password.</div>
                    <form>
                      <div class="form-floating mb-3">
                        <input class="form-control" id="inputEmail" type="email" placeholder="name@example.com" />
                        <label for="inputEmail">Email address</label>
                      </div>
                      <div class="d-flex align-items-center justify-content-between mt-4 mb-0">
                        <a class="small" href="login.html">Return to login</a>
                        <a class="btn btn-primary" href="login.html">Reset Password</a>
                      </div>
                    </form>
                  </div>
                  <div class="card-footer text-center py-3">
                    <div class="small"><a href="register.html">Need an account? Sign up</a></div>
                  </div>
                </div>
              </div>
            </div>
          </div>
        </main>
      </div>
      <div id="layoutAuthentication_footer">
        <footer class="py-4 bg-light mt-auto">
          <div class="container-fluid px-4">
            <div class="d-flex align-items-center justify-content-between small">
              <div class="text-muted">Copyright &copy; Your Website 2021</div>
              <div>
                <a href="#">Privacy Policy</a>
                &mdot;
                <a href="#">Terms & Conditions</a>
              </div>
            </div>
          </div>
        </footer>
      </div>
    </div>
  </body>
</html>
```

# register.xml

```
<!DOCTYPE html>
<html lang="en" >
  <head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
    <meta name="description" content="" />
    <meta name="author" content="" />
    <title>회원가입</title>
    <link href="/css/styles.css" rel="stylesheet" />
    <script src="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/js/all.min.js" crossorigin="anonymous"></script>
  </head>
  <body class="bg-primary">
    <div id="layoutAuthentication">
      <div id="layoutAuthentication_content">
        <main>
          <div class="container">
            <div class="row justify-content-center">
              <div class="col-lg-7">
                <div class="card shadow-lg border-0 rounded-lg mt-5">
                  <div class="card-header"><h3 class="text-center font-weight-light my-4">회원가입</h3></div>
                  <div class="card-body">
                    <form>
                      <div class="row mb-3">
                        <div class="col-md-6">
                          <div class="form-floating mb-3 mb-md-0">
                            <input class="form-control" id="inputFirstName" type="text" placeholder="Enter your first name" />
                            <label for="inputFirstName">First name</label>
                          </div>
                        </div>
                        <div class="col-md-6">
                          <div class="form-floating">
                            <input class="form-control" id="inputLastName" type="text" placeholder="Enter your last name" />
                            <label for="inputLastName">Last name</label>
                          </div>
                        </div>
                      </div>
                      <div class="form-floating mb-3">
                        <input class="form-control" id="inputEmail" type="email" placeholder="name@example.com" />
                        <label for="inputEmail">이메일을 입력하세요.</label>
                      </div>
                      <div class="row mb-3">
                        <div class="col-md-6">
                          <div class="form-floating mb-3 mb-md-0">
                            <input class="form-control" id="inputPassword" type="password" placeholder="Create a password" />
                            <label for="inputPassword">비밀번호</label>
                          </div>
                        </div>
                        <div class="col-md-6">
                          <div class="form-floating mb-3 mb-md-0">
                            <input class="form-control" id="inputPasswordConfirm" type="password" placeholder="Confirm password" />
                            <label for="inputPasswordConfirm">비밀번호 확인</label>
                          </div>
                        </div>
                      </div>
                    </form>
                    <div class="mt-4 mb-0">
                      <!--div class="d-grid"><a class="btn btn-primary btn-block" href="login.html">Create Account</a></div-->
                      <button type="button" class="btn btn-primary" id="register" style="width: 100%;>인증</button>
                    </div>
                  </div>
                </div>
              </div>
            </div>
          </div>
        </main>
      </div>
    </div>
  </body>
</html>
```



# welcome.xml

---

```
<!DOCTYPE html>
<html lang="en"
xmlns:th="http://www.thymeleaf.org" x
mlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
xmlns:sec="http://www.thymeleaf.org/extras/spring-security"
layout:decorator="board/layout/default_layout">

<head>
<title>xx병원</title>
</head>

<body id="page-top">
<div class="container-fluid" layout:fragment="content">
<br>
<br>
<h3 style="text-align:center">가입을 환영합니다.</h3>
<br>
<h5 style="text-align:center">이메일 인증이 완료 되었으니, 아래 로그인 버튼을 눌러 로그인하세요.</h5>
<br>
<div style="text-align:center">
<button sec:authorize="isAnonymous()" type="button" class="btn btn-danger"
onclick="location.href='/login/login.do'">로그인</button>
</div>
<br>
<br>
</div>
<!-- /.container-fluid -->

<th:block layout:fragment="script">
</th:block>
</body>

</html>
```

# chartjs.xml

```
<!DOCTYPE html>
<html lang="ko"
  xmlns:th="http://www.thymeleaf.org"
  xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
  layout:decorator="board/layout/default_layout">
<head>
<meta charset="UTF-8">
<title>방문환자차트</title>
<!-- <link rel="stylesheet" th:href="@{/css/style.css}"/> -->
<!-- Bootstrap CSS -->
<th:block layout:fragment="css">
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">
<link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.6.3/css/all.css">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.9.4/Chart.min.css">
</th:block>
</head>
<body>
  <div class="container" layout:fragment="content">
    <h1 class="mt-4"></h1>
    <ol class="breadcrumb mb-4">
      <li class="breadcrumb-item">방문 환자 차트</li>
    </ol>
    <div class="card mb-3">
      <div class="card-header">
        <i class="fas fa-chart-bar"></i> 환자 차트
      </div>
      <div class="card-body">
        <canvas id="myAreaChart" width="100%" height="46%"></canvas>
      </div>
      <div class="card-footer small text-muted" th:text="Updated ' + ${timestamp} ' ">Updated yesterday at 11:59 PM</div>
    </div>

    <th:block layout:fragment="script">
    <script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.9.4/Chart.min.js"></script>
    <script>
    Chart.defaults.global.defaultFontFamily = '-apple-system,system-ui,BlinkMacSystemFont,"Segoe UI",Roboto,"Helvetica Neue",Arial,sans-serif';
    Chart.defaults.global.defaultFontColor = '#292b2c';

    var config = {
      type: 'bar',
      data: {
        datasets: [{
          type: 'line',
          fill: 'false',
          label: "첫방문",
          hidden: false,
          yAxisID: 'y-axis-1',
          backgroundColor: "rgba(255,102,0,1)",
          borderColor: "rgba(255,102,0,1)",
        }, {
          type: 'line',
          fill: 'false',
          label: "재방문",
          hidden: false,
          yAxisID: 'y-axis-1',
          backgroundColor: "rgba(0,255,255,1)",
          borderColor: "rgba(0,255,255,1)",
        }, {
          type: 'bar',
          label: "긴급",
          yAxisID: 'y-axis-2',
          backgroundColor: "rgba(204,0,204,1)",
          borderColor: "rgba(204,0,204,1)",
        },
      ]},
    },
```

# chartjs.xml

---

```
options : {
  scales : {
    xAxes : [ {
      stacked: true,
      time : {
        unit : 'month'
      },
      gridLines : {
        display : true
      },
    } ],
    yAxes : [ {
      id: 'y-axis-1',
      ticks : {
        callback: function(label, index, labels) {
          return Intl.NumberFormat().format(label);
        }
      },
      gridLines : {
        display : true
      }
    }, {
      id: 'y-axis-2',
      position: 'right',
      ticks : {
        max: 100000000,
        callback: function(label, index, labels) {
          return Intl.NumberFormat().format(label);
        }
      },
      gridLines : {
        display : true
      }
    } ],
    legend : {
      display : true
    },
    tooltips: {
      mode: 'label',
      label: 'mylabel',
      callbacks: {
        label: function(tooltipItem, data) {
          return data.datasets[tooltipItem.datasetIndex].label + ": " + tooltipItem.yLabel.toString().replace(/\B(?=(\d{3})+(?!\d))/g, ","); },
      }
    }
  };

var ctx = document.getElementById('myAreaChart').getContext('2d');
window.myLine = new Chart(ctx, config);

ajaxRequest("005930");

function ajaxRequest(jongmokCode) {
$.ajax({
  url : "/chart/stockList.ajax",
  type : "GET",
  data : {
    'jongmokCode' : jongmokCode
  },
  success : function(data) {
    try {
      var dataArrDate = new Array();
      var dataArrOpen = new Array();
      var dataArrClose = new Array();
      var dataArrVolume = new Array();
```

# chartjs.xml

---

```
for (var i = 0; i < data.length; i++) {
  dataArrDate.push(data[i]['date']);
  dataArrOpen.push(data[i]['open']);
  dataArrClose.push(data[i]['close']);
  dataArrVolume.push(data[i]['volume']);
}

window.myLine.config.data.labels = dataArrDate;
window.myLine.config.data.datasets[0].data = dataArrOpen;
window.myLine.config.data.datasets[1].data = dataArrClose;
window.myLine.config.data.datasets[2].data = dataArrVolume;

window.myLine.update();
} catch (e) {
  this.error();
  return;
}
},
error : function() {
  alert("조회중 오류가 발생하였습니다.");
}
});
}

</script>
</th:block>
</body>
</html>
```

# amchart.xml

```
<!DOCTYPE html>
<html lang="ko"
  xmlns:th="http://www.thymeleaf.org"
  xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
  layout:decorator="board/layout/default_layout">
<head>
<meta charset="UTF-8">
<title>주가차트</title>
<style type="text/css">
#chartdiv {width: 100%; height: 500px;}
</style>
<!-- Bootstrap CSS -->
<th:block layout:fragment="css">
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">
<link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.6.3/css/all.css">
</th:block>
</head>
<body>
  <div class="container" layout:fragment="content">
<h1 class="mt-4"></h1>
<ol class="breadcrumb mb-4">
<li class="breadcrumb-item">원장님 전용 주식공간</a></li>
</ol>
  <div class="card mb-3">
<div class="card-header">
<i class="fas fa-chart-bar"></i> 주가 차트
</div>
<div class="card-body">
<div id="chartdiv"></div>
</div>
<div class="card-footer small text-muted" th:text="'Updated ' + ${timestamp} + 'Updated yesterday at 11:59 PM'></div>
</div>
</div>

<th:block layout:fragment="script">
<script src="//cdn.amcharts.com/lib/4/core.js"></script>
<script src="//cdn.amcharts.com/lib/4/charts.js"></script>
<script src="//cdn.amcharts.com/lib/4/themes/animated.js"></script>

<script>
// Themes begin
am4core.useTheme(am4themes_animated);

// Create chart
var chart = am4core.create("chartdiv", am4charts.XYChart);
chart.padding(0, 15, 0, 15);
chart.leftAxesContainer.layout = "vertical";
chart.dateFormatter.dateFormat = "yyyy-MM-dd";
chart.dataDateFormat = "yyyy-MM-dd";

var dateAxis = chart.xAxes.push(new am4charts.DateAxis());
dateAxis.renderer.grid.template.location = 0;
dateAxis.renderer.ticks.template.length = 8;
dateAxis.renderer.ticks.template.strokeOpacity = 0.1;
dateAxis.renderer.grid.template.disabled = true;
dateAxis.renderer.ticks.template.disabled = false;
dateAxis.renderer.ticks.template.strokeOpacity = 0.2;
dateAxis.renderer.minLabelPosition = 0.01;
dateAxis.renderer.maxLabelPosition = 0.99;
dateAxis.keepSelection = true;
dateAxis.minHeight = 30;
dateAxis.minZoomCount = 5;
dateAxis.dateFormats.setKey("month", "yyyy/MM/dd");
dateAxis.start = 0.9;
dateAxis.groupData = true;
dateAxis.skipEmptyPeriods = true;
dateAxis.tooltipDateFormat = "yyyy-MM-dd";
```



# amchart.xml

```
var valueAxis = chart.yAxes.push(new am4charts.ValueAxis());
valueAxis.tooltip.disabled = true;
valueAxis.zIndex = 1;
valueAxis.renderer.baseGrid.disabled = true;
valueAxis.height = am4core.percent(75);
valueAxis.renderer.gridContainer.background.fill = am4core.color("#000000");
valueAxis.renderer.gridContainer.background.fillOpacity = 0.05;
valueAxis.renderer.inside = true;
valueAxis.renderer.labels.template.verticalCenter = "bottom";
valueAxis.renderer.labels.template.padding(2, 2, 2, 2);
valueAxis.renderer.fontSize = "0.8em"

var series2 = chart.series.push(new am4charts.CandlestickSeries());
series2.dropFromOpenState.properties.fill = am4core.color("#3333FF");
series2.dropFromOpenState.properties.stroke = am4core.color("#3333FF");
series2.riseFromOpenState.properties.fill = am4core.color("#E71909");
series2.riseFromOpenState.properties.stroke = am4core.color("#E71909");
series2.dataFields.dateX = "date";
series2.dataFields.openValueY = "open";
series2.dataFields.valueY = "close";
series2.dataFields.lowValueY = "low";
series2.dataFields.highValueY = "high";
series2.clusters = false;
series2.tooltipText = "시가: {openValueY.value}\n고가: {highValueY.value}\n저가: {lowValueY.value}\n종가: {valueY.value}";
series2.name = "시가";
series2.defaultState.transitionDuration = 0;

var valueAxis3 = chart.yAxes.push(new am4charts.ValueAxis());
valueAxis3.tooltip.disabled = true;
valueAxis3.height = am4core.percent(25);
valueAxis3.zIndex = 3;
valueAxis3.marginTop = 30;
valueAxis3.renderer.baseGrid.disabled = true;
valueAxis3.renderer.inside = true;
valueAxis3.renderer.labels.template.verticalCenter = "bottom";
valueAxis3.renderer.labels.template.padding(2, 2, 2, 2);
valueAxis3.renderer.fontSize = "0.8em"
valueAxis3.renderer.gridContainer.background.fill = am4core.color("#000000");
valueAxis3.renderer.gridContainer.background.fillOpacity = 0.05;

var series3 = chart.series.push(new am4charts.ColumnSeries());
series3.dataFields.dateX = "date";
series3.clusters = false;
series3.dataFields.valueY = "volume";
series3.yAxis = valueAxis3;
series3.tooltipText = "거래량: {valueY.value}";
series3.name = "거래량";
series3.groupFields.valueY = "sum";
series3.defaultState.transitionDuration = 0;

chart.cursor = new am4charts.XYCursor();

var scrollbarX = new am4charts.XYChartScrollbar();

var sbSeries = chart.series.push(new am4charts.LineSeries());
sbSeries.dataFields.dateX = "date";
sbSeries.dataFields.valueY = "close";
scrollbarX.series.push(sbSeries);
sbSeries.disabled = true;
scrollbarX.marginBottom = 20;
chart.scrollbarX = scrollbarX;
scrollbarX.scrollbarChart.xAxes.getIndex(0).minHeight = undefined;

ajaxRequest("005930");
```

# amchart.xml

---

```
function ajaxRequest(jongmokCode) {
$.ajax({
    url: "/chart/stockList.ajax",
    type: "GET",
    data:{'jongmokCode': jongmokCode},
    success: function(data){

try {
var dataArr = new Array();

for(var i=0; i<data.length; i++){
dataArr.push({
    "date": data[i]['date'],
    "open": (data[i]['open'] > 0 ? data[i]['open'] : null),
    "high": (data[i]['high'] > 0 ? data[i]['high'] : null),
    "low": (data[i]['low'] > 0 ? data[i]['low'] : null),
    "close": (data[i]['close'] > 0 ? data[i]['close'] : null),
    "volume": (data[i]['volume'] > 0 ? data[i]['volume'] : null)
});
}

chart.data = dataArr;
} catch (e) {
this.error();
return;
}
    },
    error: function(){
        alert("조회중 오류가 발생하였습니다.");

        chart.data = new Array();
    }
});
}

</script>
</th:block>
</body>
</html>
```

# boardDetail.xml

```
<!DOCTYPE html>
<!DOCTYPE html>
<html lang="ko" xmlns:th="http://www.thymeleaf.org"
xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout">

  layout:decorator="/board/layout/default_layout">

  <head>
    <meta charset="UTF-8">
    <title> 환자 상세 화면 </title>
    <link rel="stylesheet" th:href="@{/css/style.css}" />
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">
  </head>

  <body>

    <div class="container" layout:fragment="content">

      <h2><div class="alert alert-warning" role="alert">
        환자 상세 화면
      </div></h2>
      <form id="frm" method="post">
        <div class="row">
          <div class="col-lg-4">
            <div class="input-group mb-1">
              <span class="input-group-text">번호</span>
              <input type="text" class="form-control" th:value="${board.boardIdx}" readonly>
            </div>
            <!--div class="input-group mb-3">
              <span class="input-group-text">작성일</span>
              <input type="text" class="form-control" th:value="${board.createdDatetime}" readonly>
            </div-->
            <div class="col-lg-4">
              <!--div class="input-group mb-1">
                <span class="input-group-text">조회수</span>
                <input type="text" class="form-control" th:value="${board.hitCnt}" readonly>
              </div-->
              <div class="input-group mb-1">
                <span class="input-group-text">등록일</span>
                <input type="text" class="form-control" th:value="${board.createdDatetime}" readonly>
              </div>
            </div>
            <div class="col-lg-4">
              <div class="input-group mb-3">
                <span class="input-group-text">수납금액</span>
                <input type="text" class="form-control" id="money" name="money" th:value="${board.money}">
              </div>
            </div>
          </div>
          <div class="row">
            <div class="col-lg-2">
              <div class="input-group mb-3">
                <span class="input-group-text">병명</span>
                <input type="text" class="form-control" id="disease" name="disease" th:value="${board.disease}">
              </div>
            </div>
            <div class="col-lg-2">
              <div class="input-group mb-3">
                <span class="input-group-text">환자이름</span>
                <input type="text" class="form-control" id="title" name="title" th:value="${board.title}">
              </div>
            </div>
            <div class="col-lg-2">
              <div class="input-group mb-3">
                <span class="input-group-text">생년월일</span>
                <input type="text" class="form-control" id="birth" name="birth" th:value="${board.birthday}">
              </div>
            </div>
          </div>
        </div>
      </form>
    </div>
  </body>
</html>
```

# boardDetail.xml

```
</div>
<div class="col-lg-3">
<div class="input-group mb-3">
<span class="input-group-text">답답의사</span>
<input type="text" class="form-control" id="doctor" name="doctor" th:value="${board.doctor}">
</div>
</div>
<div class="col-lg-3">
<div class="input-group mb-3">
<span class="input-group-text">답답간호사</span>
<input type="text" class="form-control" id="nurse" name="nurse" th:value="${board.nurse}">
</div>
</div>
<div class="mb-3">
<label class="form-label">접수 내역</label>
<textarea class="form-control" rows="5" id="contents" name="contents" th:text="${board.contents}"></textarea>
</div>
<div class="mb-3">
<label class="form-label">진료 내역</label>
<textarea class="form-control" rows="5" id="Clinic" name="Clinic" th:text="${board.Clinic}"></textarea>
</div>

<input type="hidden" th:name="${_csrf.parameterName}" th:value="${_csrf.token}"/>

<table class="board_detail">
<tbody>
<label class="form-label">환자 특징</label>
<tr th:each="list : ${replyList}">
<td th:text="${list.replyMemo}"></td>
<td th:text="${list.createdDatetime}"></td>
<td th:text="${list.creatorId}"></td>
<td><a href="/board/openReplyDetail.do?replyNo=" th:attrappend="href=${list.replyNo}"
class="btn btn-warning">수정/삭제</a></td>
</tr>
</tbody>
</table>

<input type="hidden" id="boardIdx" name="boardIdx" th:value="${board.boardIdx}">
</form>

<a href="#this" id="list" class="btn btn-warning">목록으로</a>
<a href="#this" id="edit" class="btn btn-warning">수정하기</a>
<a href="#this" id="delete" class="btn btn-warning">삭제하기</a>
<a href="/board/openReplyWrite.do?boardIdx="
th:attrappend="href=${board.boardIdx}" class="btn btn-warning">환자특징작성</a>

</div>

<th:block layout:fragment="script">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
<script type="text/javascript">
$(document).ready(function() {
$("#list").on("click", function() {
location.href = "/board/openBoardList.do";
});

$("#edit").on("click", function() {
var frm = $("#frm")[0];
frm.action = "/board/updateBoard.do";
frm.submit();
});

$("#delete").on("click", function () {
var frm = $("#frm")[0];
frm.action = "/board/deleteBoard.do";
frm.submit();
});
});
</script>
</th:block>
</body>

</html>
```

# boardList.xml

```
<!DOCTYPE html>
<html lang="ko"
xmlns:th="http://www.thymeleaf.org"
xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
layout:decorator="/board/layout/default_layout">

<head>
<meta charset="UTF-8">
<title>환자접수</title>
<!--link rel="stylesheet" th:href="@{/css/style.css}"/>
<!--link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">
<link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.6.3/css/all.css"-->
<!--link rel="stylesheet" th:href="@{/css/bootstrap.min.css}"/-->
<link rel="stylesheet" href="https://unpkg.com/bootstrap-table@1.19.1/dist/bootstrap-table.min.css">
<link href="https://unpkg.com/bootstrap-table@1.18.0/dist/extensions/page-jump-to/bootstrap-table-page-jump-to.min.css" rel="stylesheet">

</head>

<body>

<!--script src="https://cdn.jsdelivr.net/npm/jquery/dist/jquery.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"></script-->

<div class="container" layout:fragment="content">
<br>
<h2><div class="alert alert-primary" role="alert">
등록 환자 리스트
</div></h2>

<table class="table table-bordered table-sm" id="table" data-toggle="table" data-search="true"
data-strict-search="false" data-show-refresh="true" data-show-toggle="false" data-show-fullscreen="true"
data-show-columns="true" data-show-columns-toggle-all="false" data-detail-view="false" data-single-select="false"
data-click-to-select="false" data-minimum-count-columns="2" data-id-field="id" data-show-footer="false"
data-url="/board/boardList.ajax" data-sort-name="boardIdx" data-sort-order="desc" data-show-pagination-switch="true"
data-pagination="true" data-page-size="5" data-page-list="[5,10, 15, 25, 50, 100, all]" data-show-jump-to="true"
data-show-export="true" data-export-data-type="all" data-show-print="true">

<thead>
<tr>
<th data-field="boardIdx" data-sortable="true" data-formatter="viewButton" data-align="center">번호</th>

<th data-field="title" data-sortable="true" data-align="center">환자이름</th>
<th data-field="disease" data-sortable="true" data-align="center">병명</th>
<th data-field="birth" data-sortable="true" data-align="center">생년월일</th>
<th data-field="docton" data-sortable="true" data-align="center">담당의사</th>
<th data-field="nurse" data-sortable="true" data-align="center">담당간호사</th>
<!--th data-field="hitCnt" data-sortable="true" data-formatter="numberFormatter">조회수</th-->
<th data-field="createdDatetime" data-sortable="true" data-align="center">환자 등록 날짜</th>
<th data-field="money" data-sortable="true" data-formatter="numberFormatter" data-align="right">수납금액(원)</th>

</tr>
</thead>

</table>

<p align="right">
<a href="/board/openBoardWrite.do" class="btn btn-warning">환자 등록</a>
</p>
```

# boardList.xml

```
</div>

<th:block layout:fragment="script">
<script src="https://unpkg.com/bootstrap-table@1.19.1/dist/bootstrap-table.min.js"></script>
<script src="https://unpkg.com/bootstrap-table@1.18.0/dist/extensions/page-jump-to/bootstrap-table-page-jump-to.min.js"></script>
<script src="https://unpkg.com/tableexport.jquery.plugin/tableExport.min.js"></script>
<script src="https://unpkg.com/bootstrap-table@1.18.0/dist/extensions/export/bootstrap-table-export.min.js"></script>
<script src="https://unpkg.com/bootstrap-table@1.18.0/dist/extensions/print/bootstrap-table-print.min.js"></script>

<script>

function viewButton(value) {
return '<button type="button" class="btn btn-secondary btn-sm" style="padding-top:0">' + value +
'</button>'
};

function numberFormatter(value) {
if (value === 0) return 0;
const reg = /^(?=[-]?)(\d{3})/;
let n = (value + '');
while (reg.test(n)) n = n.replace(reg, '$1' + ',' + '$2');
return n;
}

$("#table").on("load-success.bs.table", function (data) {
var data = $("#table").bootstrapTable("getData")
if (data.length > 0) {
//alert(data[0].boardIdx);
}
});

$("#table").on("click", "tr", function () {
var row = $(this).index();
var tr = $(this);
var td = tr.children();
var id = td.eq(0).text();

if ($.isNumeric(id)) {
location.href = "/board/openBoardDetail.do?boardIdx=" + id;
}
});
</script>
</th:block>

</body>
</html>
```

# boardWrite.xml

```
<!DOCTYPE html>
<html lang="ko"
xmlns:th="http://www.thymeleaf.org"
xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
layout:decorator="/board/layout/default_layout">

<head>
<meta charset="UTF-8">
<title>환자 등록</title>
<link rel="stylesheet" th:href="@{/css/style.css}"/>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">
</head>

<body>

<div class="container" layout:fragment="content">
<h2><div class="alert alert-warning" role="alert">
환자 등록
</div></h2>
<form id="frm" name="frm" method="post" action="/board/insertBoard.do">
<div class="input-group mb-1">
<span class="input-group-text">이 름</span>
<input type="text" class="form-control" id="title" name="title">
</div>

<div class="input-group mb-1">
<span class="input-group-text">병 명</span>
<input type="text" class="form-control" id="disease" name="disease">
</div>
<div class="input-group mb-1">
<span class="input-group-text">생년월일</span>
<input type="text" class="form-control" id="birth" name="birth">
</div>
<div class="input-group mb-1">
<span class="input-group-text">담당의사</span>
<input type="text" class="form-control" id="doctor" name="doctor">
</div>
<div class="input-group mb-1">
<span class="input-group-text">담당간호사</span>
<input type="text" class="form-control" id="nurse" name="nurse">
</div>
<div class="input-group mb-1">
<span class="input-group-text">수납금액</span>
<input type="text" class="form-control" id="money" name="money">
</div>

<div class="mb-3">
<label class="form-label">접수 내역</label>
<textarea class="form-control" rows="5" id="contents" name="contents"></textarea>
</div>
<div class="mb-3">
<label class="form-label">진료 내역</label>
<textarea class="form-control" rows="5" id="Clinic" name="Clinic"></textarea>
</div>

<input type="hidden" th:name="${_csrf.parameterName}" th:value="${_csrf.token}"/>
<p align="right">
<input type="submit" id="submit" value="저장" class="btn btn-warning">
</p>
</form>
</div>
</body>
<th:block layout:fragment="script">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
</th:block>
</html>
```

# replyDetail.xml

```
<!DOCTYPE html>
<!DOCTYPE html>
<html lang="ko" xmlns:th="http://www.thymeleaf.org">
<head>
<meta charset="UTF-8">
<title> 한자 특징 등록 </title>
<link rel="stylesheet" th:href="@{/css/style.css}" />
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">

</head>

<body>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
<div class="container">
<h2><div class="alert alert-warning" role="alert">
한자 특징
</div></h2>
<form id="frm" method="post">
<table class="board_detail">
<colgroup>
<col width="15%" />
<col width="35%" />
<col width="15%" />
<col width="35%" />
</colgroup>
<caption> 한자 상세내용</caption>
<tbody>
<tr>
<th scope="row"> 번호</th>
<td th:text="${board.replyNo}"></td>
<tr>
<th scope="row"> 작성자</th>
<td th:text="${board.creatorId}"></td>
<th scope="row"> 작성일</th>
<td th:text="${board.createdDatetime}"></td>
</tr>
<tr>
<th scope="row"> 특징</th>
<td colspan="3"><input type="text" id="replyMemo" name="replyMemo" th:value="${board.replyMemo}" /></td>
</tr>
</tbody>
</table>

<input type="hidden" id="replyNo" name="replyNo" th:value="${board.replyNo}">
<input type="hidden" th:name="${_csrf.parameterName}" th:value="${_csrf.token}" />
</form>

<a href="/board/openBoardDetail.do?boardIdx="
th:attrappend="href=${board.boardIdx}" class="btn btn-warning">한자 정보 상세</a>
<a href="#this" id="edit" class="btn btn-warning">수정하기</a>
<a href="#this" id="delete" class="btn btn-warning">삭제하기</a>

</div>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
<script type="text/javascript">
$(document).ready(function() {

$("#edit").on("click", function() {
var frm = $("#frm")[0];
frm.action = "/board/updateReply.do";
frm.submit();

});
$("#delete").on("click", function () {
var frm = $("#frm")[0];
frm.action = "/board/deleteReply.do";
frm.submit();
});

});
</script>
</body>
```



# replyWrite.xml

---

```
<!DOCTYPE html>
<html lang="ko" xmlns:th="http://www.thymeleaf.org">

<head>
<meta charset="UTF-8">
<title>특징 등록</title>
<link rel="stylesheet" th:href="@{/css/style.css}"/>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">
</head>

<body>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
<div class="container">
<h2>환자 특징 등록</h2>
<form id="frm" name="frm" method="post" action="/board/insertReply.do">
<table class="board_detail">
<tr>
<td colspan="2">
<textarea id="replyMemo" name="replyMemo"></textarea>
</td>
</tr>
</table>
<input type="hidden" id="boardIdx" name="boardIdx"
th:value="${boardIdx}">
<input type="hidden" th:name="${_csrf.parameterName}" th:value="${_csrf.token}"/>
<p align="right">
<input type="submit" id="submit" value="저장" class="btn btn-warning">
</p>
</form>

</div>

</body>

</html>
```

# staffDetail.xml

```
<!DOCTYPE html>
<!DOCTYPE html>
<html lang="ko" xmlns:th="http://www.thymeleaf.org"
xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout">

    layout:decorator="/board/layout/default_layout">

<head>
<meta charset="UTF-8">
<title> 직원 상세 화면 </title>
<link rel="stylesheet" th:href="@{/css/style.css}" />
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">
</head>

<body>

<div class="container" layout:fragment="content">

<h2><div class="alert alert-warning" role="alert">
직원 상세 화면
</div></h2>
<form id="frm" method="post">
<div class="row">
<div class="col-lg-4">
<div class="input-group mb-1">
<span class="input-group-text">번호</span>
<input type="text" class="form-control" th:value="${board.staffNo}" readonly>
</div>
<!--div class="input-group mb-3">
<span class="input-group-text">작성일</span>
<input type="text" class="form-control" th:value="${board.creatorId}" readonly>
</div-->
</div>
<div class="col-lg-4">
<!--div class="input-group mb-1">
<span class="input-group-text">조회수</span>
<input type="text" class="form-control" th:value="${board.hitCnt}" readonly>
</div-->
<div class="input-group mb-1">
<span class="input-group-text">등록일</span>
<input type="text" class="form-control" th:value="${board.createdDatetime}" readonly>
</div>
<div class="col-lg-4">
<div class="input-group mb-1">
<span class="input-group-text">수정일</span>
<input type="text" class="form-control" th:value="${board.createdDatetime}" readonly>
</div>
</div>
</div>
<div class="row">
<div class="col-lg-3">
<div class="input-group mb-3">
<span class="input-group-text">직원이름</span>
<input type="text" class="form-control" id="name" name="name" th:value="${board.name}">
</div>
</div>
<div class="col-lg-3">
<div class="input-group mb-3">
<span class="input-group-text">생년월일</span>
<input type="text" class="form-control" id="birth" name="birth" th:value="${board.birth}">
</div>
</div>
<div class="col-lg-3">
<div class="input-group mb-3">
<span class="input-group-text">전화번호</span>
<input type="text" class="form-control" id="Phone" name="Phone" th:value="${board.Phone}">
</div>
</div>
</div>
</div>
```

# staffDetail.xml

```
</div>
<div class="col-lg-3">
<div class="input-group mb-3">
<span class="input-group-text">입사일자</span>
<input type="text" class="form-control" id="date" name="date" th:value="${board.date}">
</div>
</div>
</div>
<div class="mb-3">
<label class="form-label">연봉</label>
<textarea class="form-control" rows="5" id="contents" name="contents" th:text="${board.contents}"></textarea>
</div>

<input type="hidden" th:name="${_csrf.parameterName}" th:value="${_csrf.token}"/>

<table class="board_detail">

</table>

<div>

<input type="hidden" id="staffNo" name="staffNo" th:value="${board.staffNo}">
</form>

<a href="#this" id="list" class="btn btn-warning">목록으로</a>
<a href="#this" id="edit" class="btn btn-warning">수정하기</a>
<a href="#this" id="delete" class="btn btn-warning">삭제하기</a>

</div>

<th:block layout:fragment="script">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
<script type="text/javascript">
$(document).ready(function() {
$("#list").on("click", function() {
location.href = "/board/openStaffList.do";
});

$("#edit").on("click", function() {
var frm = $("#frm")[0];
frm.action = "/board/updateStaff.do";
frm.submit();
});

$("#delete").on("click", function () {
var frm = $("#frm")[0];
frm.action = "/board/deleteStaff.do";
frm.submit();
});
});
</script>
</th:block>
</body>

</html>
```

# staffList.xml

```
<!DOCTYPE html>
<html lang="ko"
xmlns:th="http://www.thymeleaf.org"
xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
layout:decorator="/board/layout/default_layout">

<head>
<meta charset="UTF-8">
<title>직원 리스트</title>
<!--link rel="stylesheet" th:href="@{/css/style.css}"/>
<!--link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">
<link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.6.3/css/all.css"-->
<!--link rel="stylesheet" th:href="@{/css/bootstrap.min.css}"/-->
<link rel="stylesheet" href="https://unpkg.com/bootstrap-table@1.19.1/dist/bootstrap-table.min.css">
<link href="https://unpkg.com/bootstrap-table@1.18.0/dist/extensions/page-jump-to/bootstrap-table-page-jump-to.min.css" rel="stylesheet">

</head>

<body>

<!--script src="https://cdn.jsdelivr.net/npm/jquery/dist/jquery.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"></script-->

<div class="container" layout:fragment="content">
<br>
<h2><div class="alert alert-primary" role="alert">
직원 리스트
</div></h2>

<table class="table table-bordered table-sm" id="table" data-toggle="table" data-search="true"
data-strict-search="false" data-show-refresh="true" data-show-toggle="false" data-show-fullscreen="true"
data-show-columns="true" data-show-columns-toggle-all="false" data-detail-view="false" data-single-select="false"
data-click-to-select="false" data-minimum-count-columns="2" data-id-field="id" data-show-footer="false"
data-url="/board/staffList.ajax" data-sort-name="staffNo" data-sort-order="desc" data-show-pagination-switch="true"
data-pagination="true" data-page-size="5" data-page-list="[5,10, 15, 25, 50, 100, all]" data-show-jump-to="true"
data-show-export="true" data-export-data-type="all" data-show-print="true">

<thead>

<tr>
<th data-field="staffNo" data-sortable="true" data-formatter="viewButton" data-align="center">번호</th>
<th data-field="name" data-sortable="true" data-align="center">직원이름</th>
<th data-field="birth" data-sortable="true" data-align="center">생년월일</th>
<th data-field="phone" data-sortable="true" data-align="center">전화번호</th>
<th data-field="date" data-sortable="true" data-align="center">입사일</th>
<!--th data-field="hitCnt" data-sortable="true" data-formatter="numberFormatter">조회수</th-->
<th data-field="createdDatetime" data-sortable="true" data-align="center">직원 등록 날짜</th>
<!--th data-field="updatedDatetime" data-sortable="true" data-align="center">마지막 방문날짜</th-->

</tr>
</thead>

</table>

<p align="right">
<a href="/board/openStaffWrite.do" class="btn btn-warning">직원 등록</a>
</p>
</div>
```

# staffList.xml

---

```
<th:block layout:fragment="script">
<script src="https://unpkg.com/bootstrap-table@1.19.1/dist/bootstrap-table.min.js"></script>
<script src="https://unpkg.com/bootstrap-table@1.18.0/dist/extensions/page-jump-to/bootstrap-table-page-jump-to.min.js"></script>
<script src="https://unpkg.com/tableexport.jquery.plugin/tableExport.min.js"></script>
<script src="https://unpkg.com/bootstrap-table@1.18.0/dist/extensions/export/bootstrap-table-export.min.js"></script>
<script src="https://unpkg.com/bootstrap-table@1.18.0/dist/extensions/print/bootstrap-table-print.min.js"></script>

</script>

function viewButton(value) {
return 'button type="button" class="btn btn-secondary btn-sm" style="padding-top:0">' + value +
'</button>'
};

function numberFormatter(value) {
if (value === 0) return 0;
const reg = /^(+)?\d+)(\d{3})/;
let n = (value + '');
while (reg.test(n)) n = n.replace(reg, '$1' + ',' + '$2');
return n;
}

$("#table").on("load-success.bs.table", function (data) {
var data = $("#table").bootstrapTable("getData")
if (data.length > 0) {
//alert(data[0].boardIdx);
}
});

$("#table").on("click", "tr", function () {
var row = $(this).index();
var tr = $(this);
var td = tr.children();
var id = td.eq(0).text();

if ($.isNumeric(id)) {
location.href = "/board/openStaffDetail.do?staffNo=" + id;
}
});
</script>
</th:block>
</body>
</html>
```

# staffwrite.xml

```
<!DOCTYPE html>
<html lang="ko"
xmlns:th="http://www.thymeleaf.org"
xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
layout:decorator="/board/layout/default_layout">

<head>
<meta charset="UTF-8">
<title>직원 등록</title>
<link rel="stylesheet" th:href="@{/css/style.css}"/>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">
</head>

<body>

<div class="container" layout:fragment="content">
<h2><div class="alert alert-warning" role="alert">
직원 등록
</div></h2>

<form id="frm" name="frm" method="post" action="/board/insertStaff.do">
<div class="row">
<div class="col-lg-3">
<div class="input-group mb-1">
<span class="input-group-text">이름</span>
<input type="text" class="form-control" id="name" name="name">
</div>
</div>
<div class="col-lg-3">
<div class="input-group mb-1">
<span class="input-group-text">생년월일</span>
<input type="text" class="form-control" id="birth" name="birth">
</div>
</div>
<div class="col-lg-3">
<div class="input-group mb-1">
<span class="input-group-text">전화번호</span>
<input type="text" class="form-control" id="Phone" name="Phone">
</div>
</div>
<div class="col-lg-3">
<div class="input-group mb-1">
<span class="input-group-text">입사일자</span>
<input type="text" class="form-control" id="date" name="date">
</div>
</div>
</div>

<div class="mb-3">
<label class="form-label">연봉</label>
<textarea class="form-control" rows="3" id="contents" name="contents"></textarea>
</div>

<input type="hidden" th:name="${_csrf.parameterName}" th:value="${_csrf.token}"/>
<p align="right">
<input type="submit" id="submit" value="저장" class="btn btn-warning">
</p>
</form>
</div>
</body>
<th:block layout:fragment="script">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
</th:block>
</html>
```

**감사합니다.**