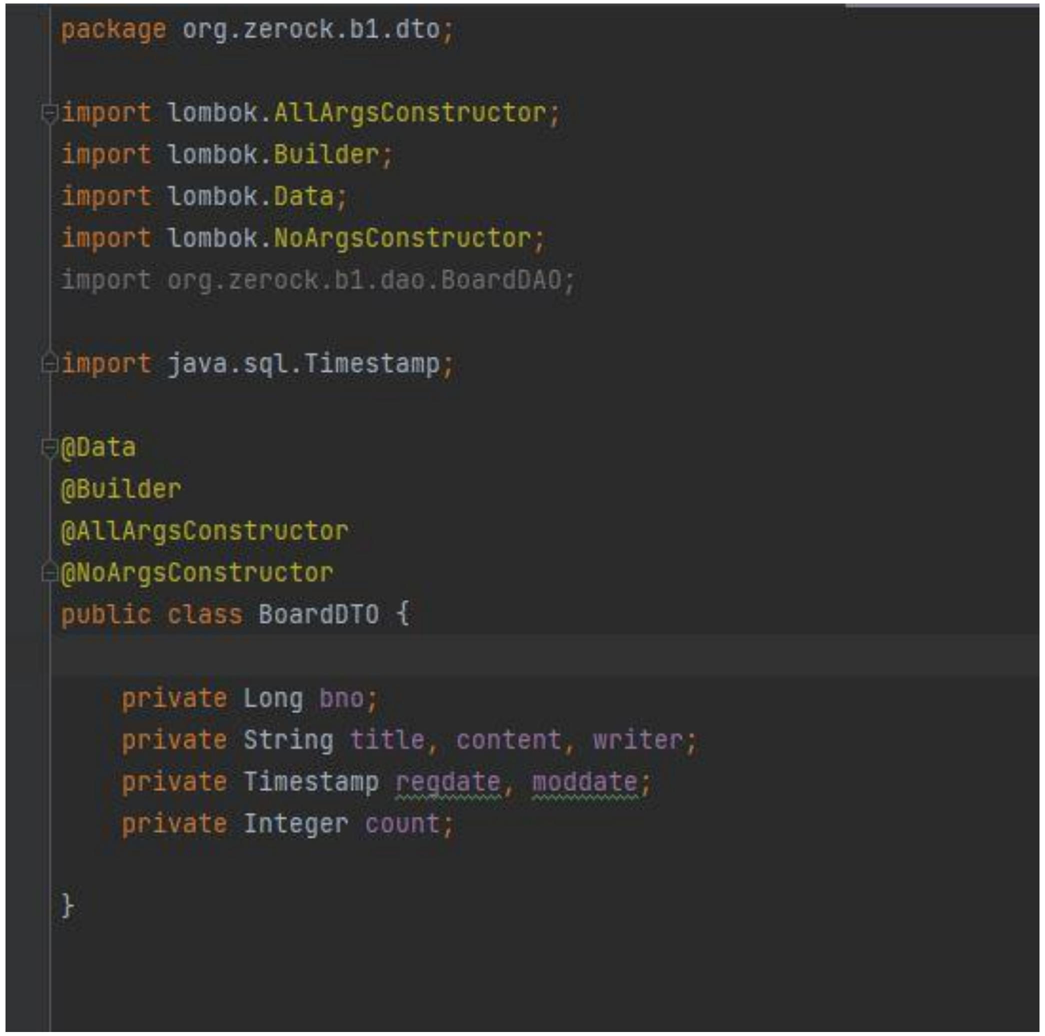
# 게시판 만들기

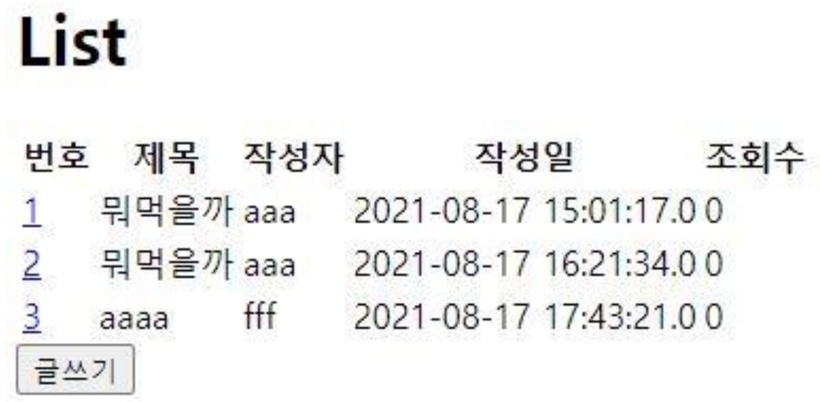
남승범, 이아현

# BoardDTO



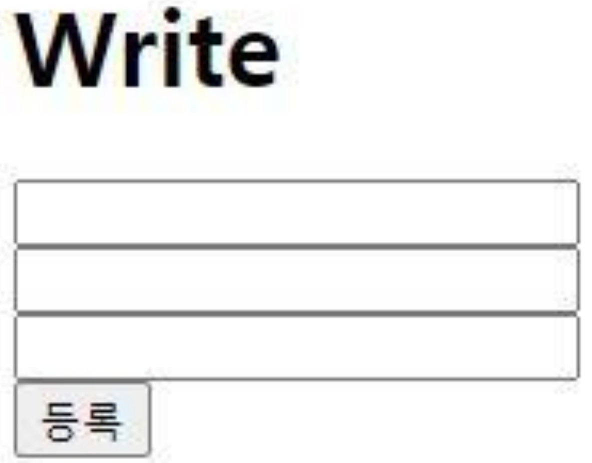
```java
package org.zerock.b1.dto;

import lombok.AllArgsConstructor;
import lombok.Builder;
import lombok.Data;
import lombok.NoArgsConstructor;
import org.zerock.b1.dao.BoardDAO;

import java.sql.Timestamp;

@Data
@Builder
@AllArgsConstructor
@NoArgsConstructor
public class BoardDTO {

    private Long bno;
    private String title, content, writer;
    private Timestamp regdate, moddate;
    private Integer count;

}
```

# 화면구현



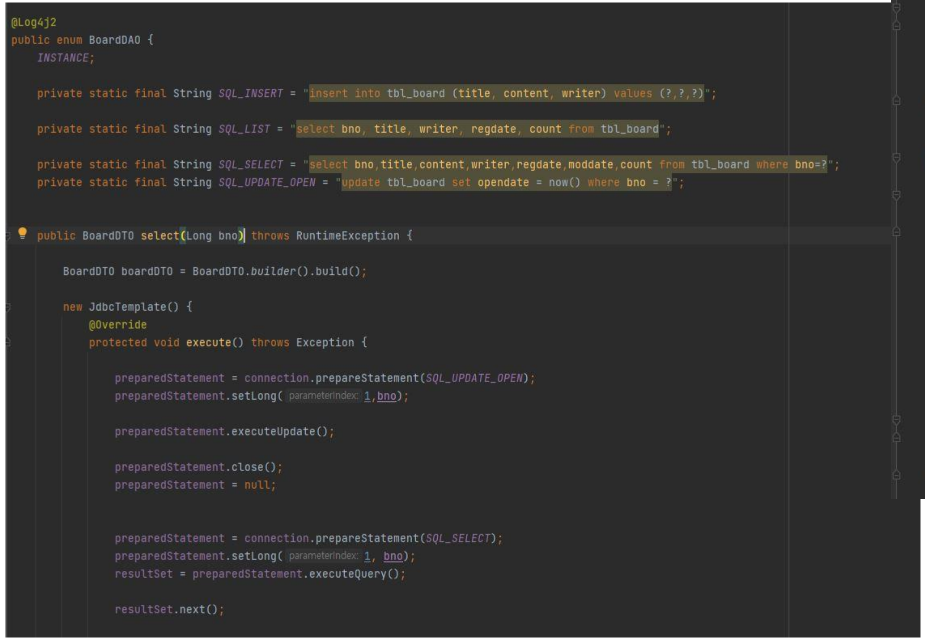
## List

| 번호 | 제목 | 작성자 | 작성일 | 조회수 |
|---|---|---|---|---|
| 1 | 뭐먹을까 | aaa | 2021-08-17 15:01:17.0 | 0 |
| 2 | 뭐먹을까 | aaa | 2021-08-17 16:21:34.0 | 0 |
| 3 | aaaa | fff | 2021-08-17 17:43:21.0 | 0 |

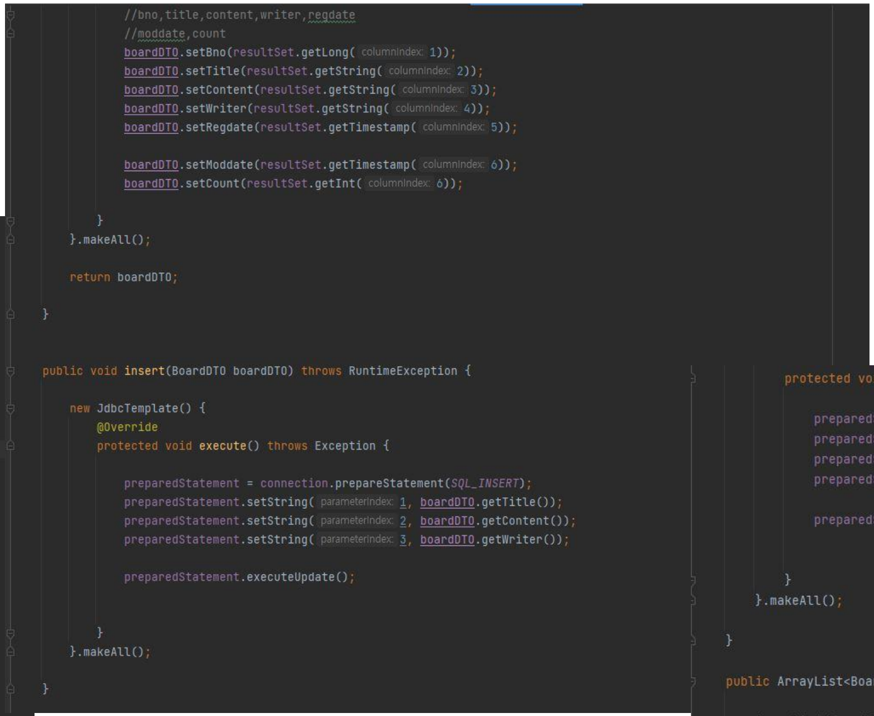


## Write

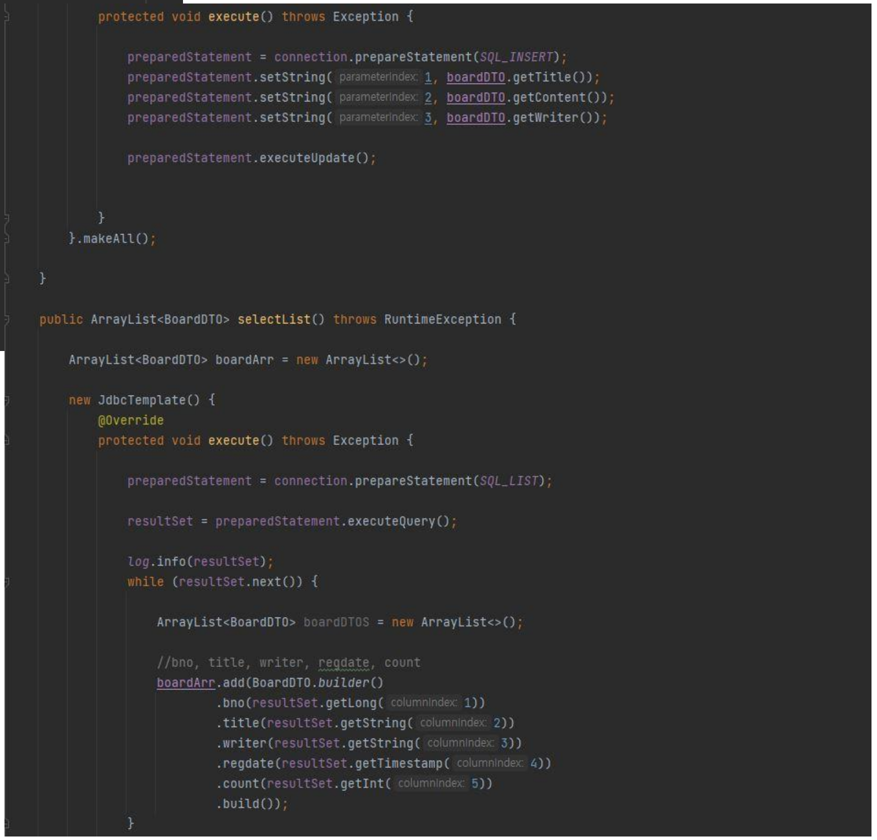
# BoardDAO



```java
@Log4j2
public enum BoardDAO {
    INSTANCE;

    private static final String SQL_INSERT = "insert into tbl_board (title, content, writer) values (?,?,?)";

    private static final String SQL_LIST = "select bno, title, writer, regdate, count from tbl_board";

    private static final String SQL_SELECT = "select bno,title,content,writer,regdate,moddate,count from tbl_board where bno=?";
    private static final String SQL_UPDATE_OPEN = "update tbl_board set opendate = now() where bno = ?";


    public BoardDTO select(Long bno) throws RuntimeException {

        BoardDTO boardDTO = BoardDTO.builder().build();

        new JdbcTemplate() {
            @Override
            protected void execute() throws Exception {

                preparedStatement = connection.prepareStatement(SQL_UPDATE_OPEN);
                preparedStatement.setLong( parameterIndex: 1,bno);

                preparedStatement.executeUpdate();

                preparedStatement.close();
                preparedStatement = null;


                preparedStatement = connection.prepareStatement(SQL_SELECT);
                preparedStatement.setLong( parameterIndex: 1, bno);
                resultSet = preparedStatement.executeQuery();

                resultSet.next();
```



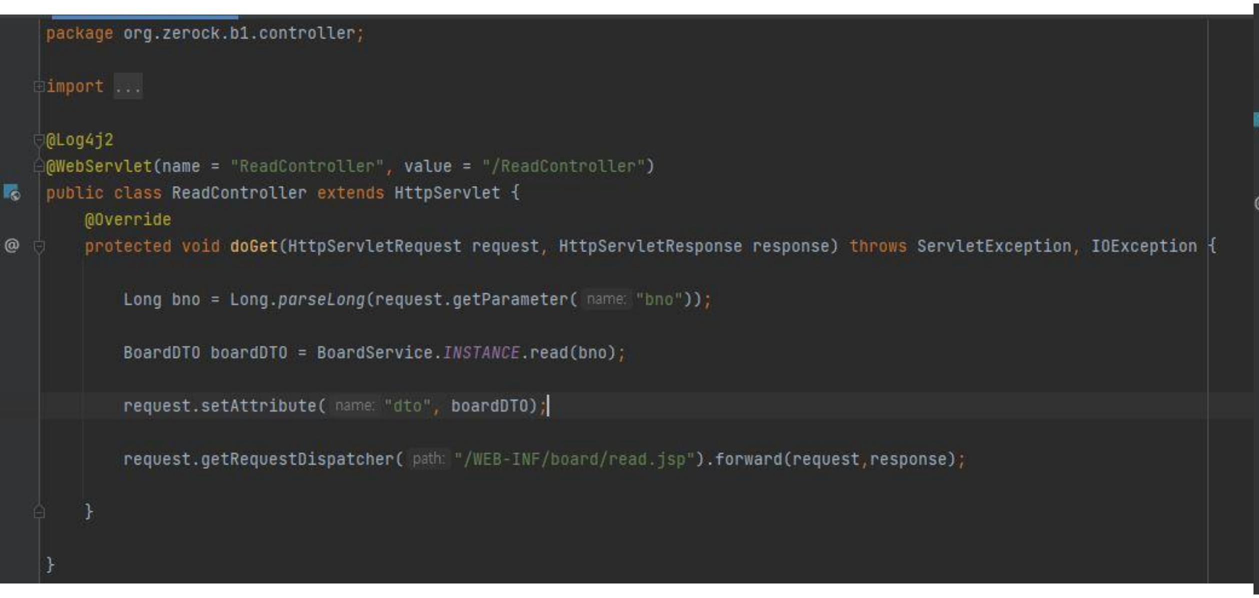
```java
                //bno,title,content,writer,regdate
                //moddate,count
                boardDTO.setBno(resultSet.getLong( columnIndex: 1));
                boardDTO.setTitle(resultSet.getString( columnIndex: 2));
                boardDTO.setContent(resultSet.getString( columnIndex: 3));
                boardDTO.setWriter(resultSet.getString( columnIndex: 4));
                boardDTO.setRegdate(resultSet.getTimestamp( columnIndex: 5));

                boardDTO.setModdate(resultSet.getTimestamp( columnIndex: 6));
                boardDTO.setCount(resultSet.getInt( columnIndex: 6));

            }
        }.makeAll();

        return boardDTO;

    }


    public void insert(BoardDTO boardDTO) throws RuntimeException {

        new JdbcTemplate() {
            @Override
            protected void execute() throws Exception {

                preparedStatement = connection.prepareStatement(SQL_INSERT);
                preparedStatement.setString( parameterIndex: 1, boardDTO.getTitle());
                preparedStatement.setString( parameterIndex: 2, boardDTO.getContent());
                preparedStatement.setString( parameterIndex: 3, boardDTO.getWriter());

                preparedStatement.executeUpdate();


            }
        }.makeAll();

    }
```



```java
            protected void execute() throws Exception {

                preparedStatement = connection.prepareStatement(SQL_INSERT);
                preparedStatement.setString( parameterIndex: 1, boardDTO.getTitle());
                preparedStatement.setString( parameterIndex: 2, boardDTO.getContent());
                preparedStatement.setString( parameterIndex: 3, boardDTO.getWriter());

                preparedStatement.executeUpdate();


            }
        }.makeAll();

    }

    public ArrayList<BoardDTO> selectList() throws RuntimeException {

        ArrayList<BoardDTO> boardArr = new ArrayList<>();

        new JdbcTemplate() {
            @Override
            protected void execute() throws Exception {

                preparedStatement = connection.prepareStatement(SQL_LIST);

                resultSet = preparedStatement.executeQuery();

                log.info(resultSet);
                while (resultSet.next()) {

                    ArrayList<BoardDTO> boardDTOS = new ArrayList<>();

                    //bno, title, writer, regdate, count
                    boardArr.add(BoardDTO.builder()
                            .bno(resultSet.getLong( columnIndex: 1))
                            .title(resultSet.getString( columnIndex: 2))
                            .writer(resultSet.getString( columnIndex: 3))
                            .regdate(resultSet.getTimestamp( columnIndex: 4))
                            .count(resultSet.getInt( columnIndex: 5))
                            .build());
                }
```

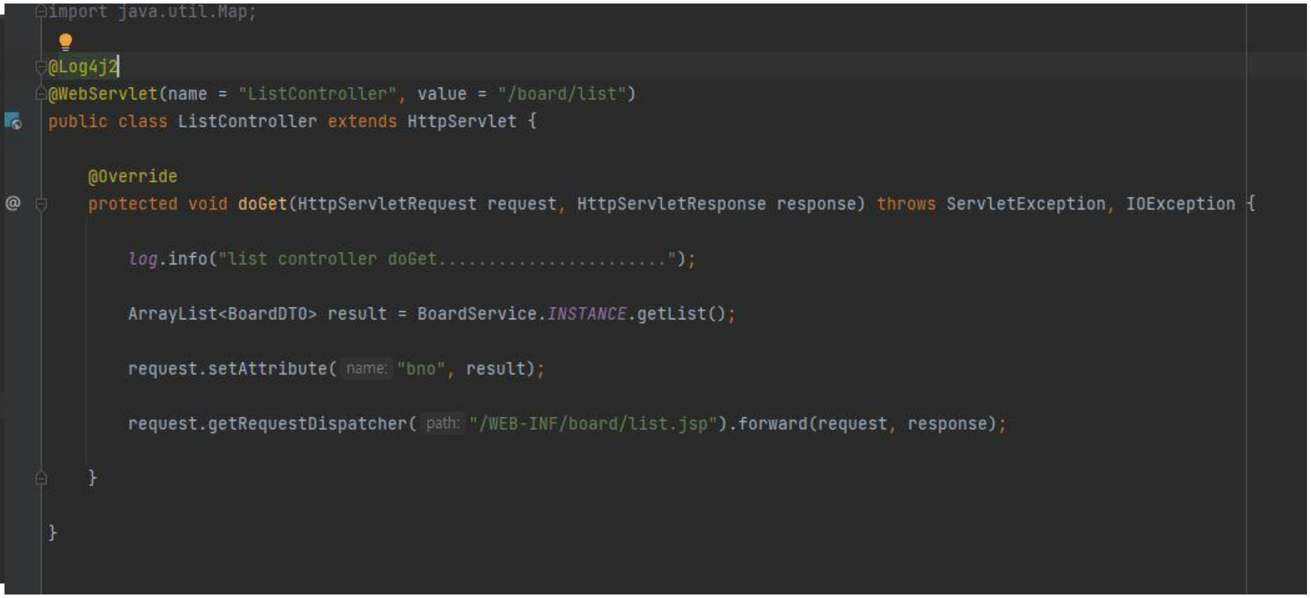
# controller

ReadController



```
package org.zerock.b1.controller;

import ...

@Log4j2
@WebServlet(name = "ReadController", value = "/ReadController")
public class ReadController extends HttpServlet {
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

        Long bno = Long.parseLong(request.getParameter( name: "bno"));

        BoardDTO boardDTO = BoardService.INSTANCE.read(bno);

        request.setAttribute( name: "dto", boardDTO);

        request.getRequestDispatcher( path: "/WEB-INF/board/read.jsp").forward(request,response);

    }

}
```

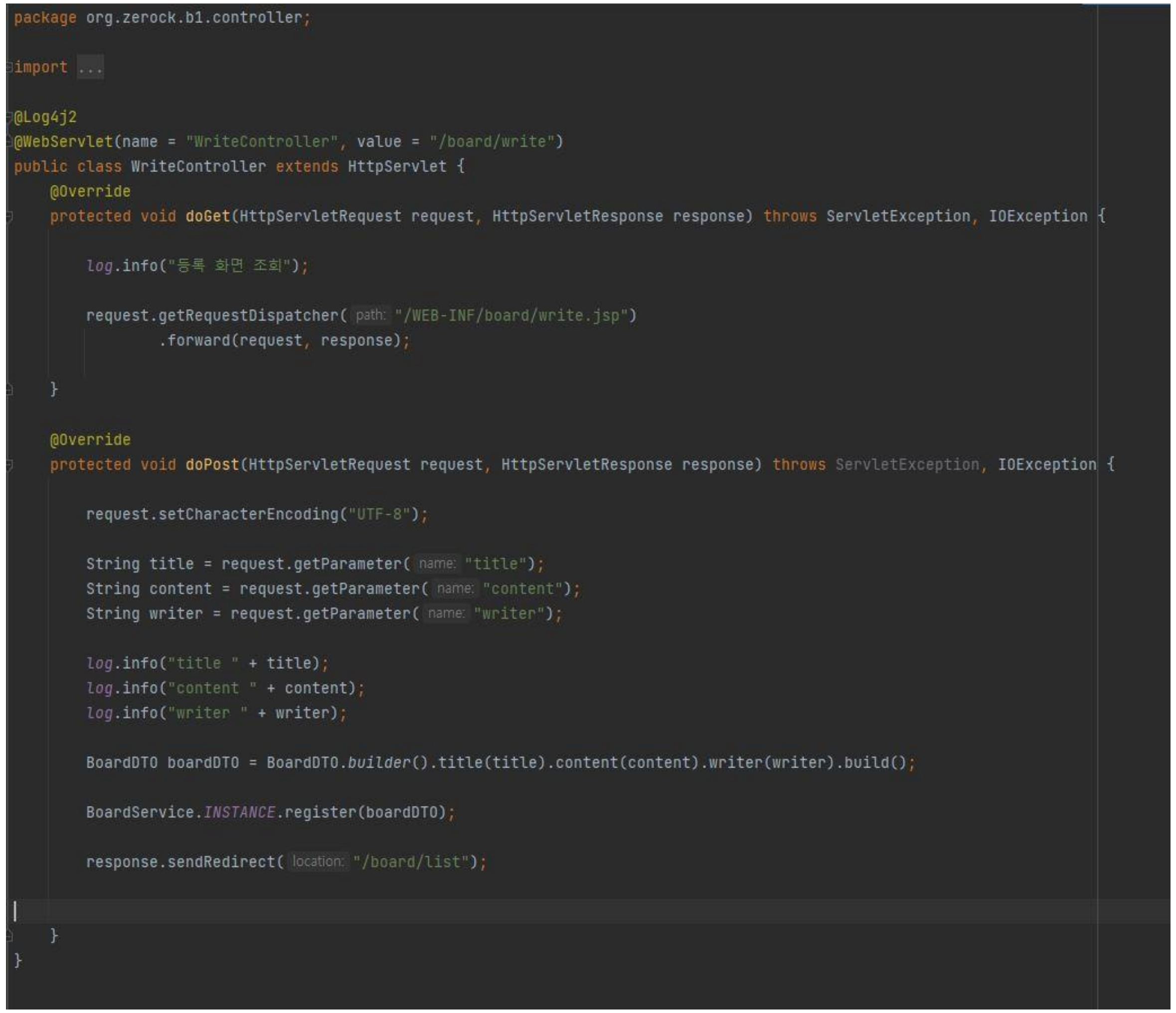
ListController



```
import java.util.Map;

@Log4j2
@WebServlet(name = "ListController", value = "/board/list")
public class ListController extends HttpServlet {

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

        log.info("list controller doGet......................");

        ArrayList<BoardDTO> result = BoardService.INSTANCE.getList();

        request.setAttribute( name: "bno", result);

        request.getRequestDispatcher( path: "/WEB-INF/board/list.jsp").forward(request, response);

    }

}
```

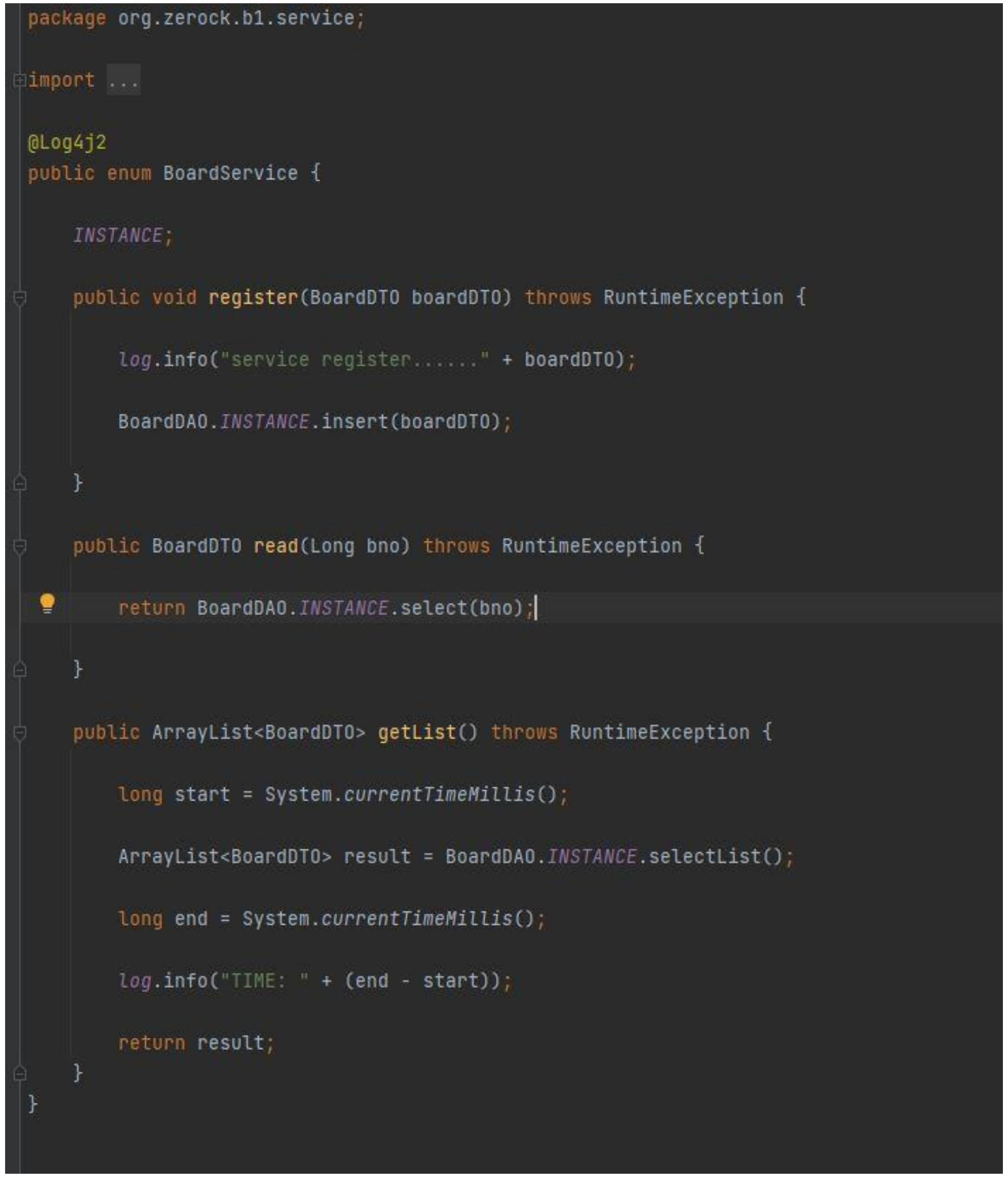
# controller

WriteController



```java
package org.zerock.b1.controller;

import ...

@Log4j2
@WebServlet(name = "WriteController", value = "/board/write")
public class WriteController extends HttpServlet {
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

        log.info("등록 화면 조회");

        request.getRequestDispatcher( path: "/WEB-INF/board/write.jsp")
                .forward(request, response);

    }

    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

        request.setCharacterEncoding("UTF-8");

        String title = request.getParameter( name: "title");
        String content = request.getParameter( name: "content");
        String writer = request.getParameter( name: "writer");

        log.info("title " + title);
        log.info("content " + content);
        log.info("writer " + writer);

        BoardDTO boardDTO = BoardDTO.builder().title(title).content(content).writer(writer).build();

        BoardService.INSTANCE.register(boardDTO);

        response.sendRedirect( location: "/board/list");

    }
}
```

## service



```
package org.zerock.b1.service;

import ...

@Log4j2
public enum BoardService {

    INSTANCE;

    public void register(BoardDTO boardDTO) throws RuntimeException {

        log.info("service register......" + boardDTO);

        BoardDAO.INSTANCE.insert(boardDTO);

    }

    public BoardDTO read(Long bno) throws RuntimeException {

        return BoardDAO.INSTANCE.select(bno);

    }

    public ArrayList<BoardDTO> getList() throws RuntimeException {

        long start = System.currentTimeMillis();

        ArrayList<BoardDTO> result = BoardDAO.INSTANCE.selectList();

        long end = System.currentTimeMillis();

        log.info("TIME: " + (end - start));

        return result;
    }
}
```