

# REPORT

애플리케이션 설계  
정기 수행평가

과 목 명	애플리케이션 설계
능 력 단 위	공통 모듈 설계
성 명	장경준

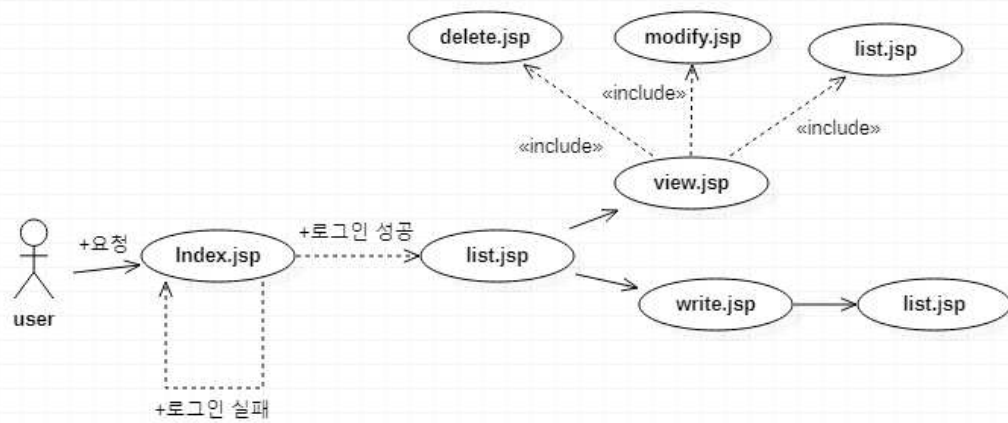
수행평가 실습 보고서			
능력단위 요소	공통 모듈 설계	작 성 자	장경준
평가방법	서술형	작성일자	
단 계 명	구현	문서번호	#1

## 1.SDK(System Development Kit) 또는 개발 도구 정보

구분	내용
SDK(개발도구) 명칭	eclipse
SDK(개발도구) 버전	eclipse 2021-03
SDK(개발도구)설명 (주요기능의 설명)	<ul style="list-style-type: none"> <li>● 여러 프로젝트 동시 관리 가능.</li> <li>● 자바로 만들어져 자바 가상머신 위에서 실행.</li> </ul>
SDK(개발도구) 사용방법 (가격, 절차 등)	오픈소스

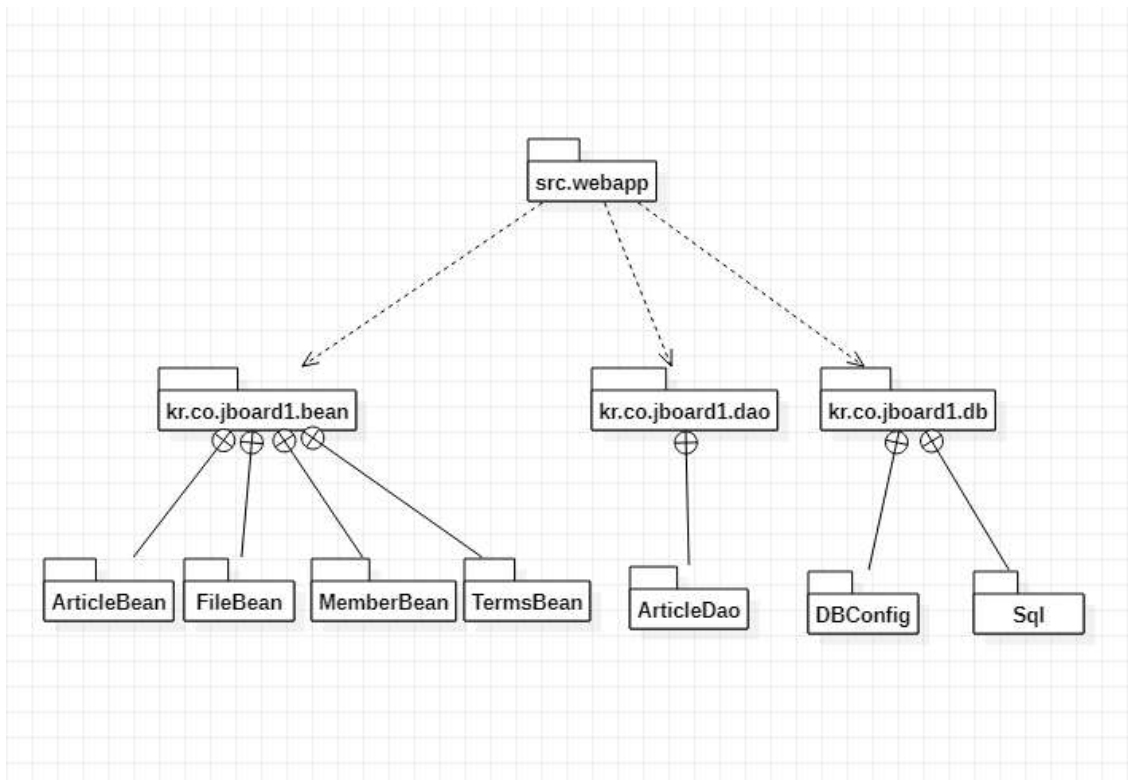
## 2. 평가문항 풀이

## 1. 게시판 애플리케이션 모듈 설계



< StarUML 로 작성 >

## 2. 공통 모듈 클래스 설계 및 구현



### 3. 공통 모듈 클래스 코드 첨부 및 설명

#### 3-1. bean 패키지 설계

: 객체를 관리하기 위한 생성자 설계

```
1 package kr.co.jboard1.bean;
2
3 public class ArticleBean {
4
5     private int seq;
6     private int parent;
7     private int comment;
8     private String cate;
9     private String title;
10    private String content;
11    private int file;
12    private int hit;
13    private String uid;
14    private String regip;
15    private String rdate;
16
17    | private String nick;
18    private FileBean fb;
19
20    public FileBean getFb() {
21        return fb;
22    }
23    public void setFb(FileBean fb) {
24        this.fb = fb;
25    }
26
27    public String getNick() {
28        return nick;
29    }
30    public void setNick(String nick) {
31        this.nick = nick;
32    }
33
34    public int getSeq() {
35        return seq;
36    }
37    public void setSeq(int seq) {
38        this.seq = seq;
39    }
40    public int getParent() {
41        return parent;
42    }
43
44    public void setParent(int parent) {
45        this.parent = parent;
46    }
47    public void setParent(String parent) {
48        this.parent = Integer.parseInt(parent);
49    }
50 }
```

< ArticleBean.java>

```

1 package kr.co.jboard1.bean;
2
3 public class FileBean {
4
5     private int seq;
6     private int parent;
7     private String oriName;
8     private String newName;
9     private int download;
10    private String rdate;
11
12    public int getSeq() {
13        return seq;
14    }
15    public void setSeq(int seq) {
16        this.seq = seq;
17    }
18    public int getParent() {
19        return parent;
20    }
21    public void setParent(int parent) {
22        this.parent = parent;
23    }
24    public String getOriName() {
25        return oriName;
26    }
27    public void setOriName(String oriName) {
28        this.oriName = oriName;
29    }
30    public String getNewName() {
31        return newName;
32    }
33    public void setNewName(String newName) {
34        this.newName = newName;
35    }
36    public int getDownload() {
37        return download;
38    }
39    public void setDownload(int download) {
40        this.download = download;
41    }
42    public String getRdate() {
43        return rdate;
44    }
45    public void setRdate(String rdate) {
46        this.rdate = rdate;
47    }
48 }

```

< FileBean.java >



```
1 package kr.co.jboard1.bean;
2
3 public class MemberBean {
4     private String uid;
5     private String pass;
6     private String name;
7     private String nick;
8     private String email;
9     private String hp;
10    private int grade;
11    private String zip;
12    private String addr1;
13    private String addr2;
14    private String regip;
15    private String rdate;
16
17    public String getUid() {
18        return uid;
19    }
20    public void setUid(String uid) {
21        this.uid = uid;
22    }
23    public String getPass() {
24        return pass;
25    }
26    public void setPass(String pass) {
27        this.pass = pass;
28    }
29    public String getName() {
30        return name;
31    }
32    public void setName(String name) {
33        this.name = name;
34    }
35    public String getNick() {
36        return nick;
37    }
38    public void setNick(String nick) {
39        this.nick = nick;
40    }
41    public String getEmail() {
42        return email;
43    }
44    public void setEmail(String email) {
45        this.email = email;
46    }
47    public String getHp() {
48        return hp;
49    }
50 }
```

< MemberBean.java >

```

1 package kr.co.jboard1.bean;
2
3 public class TermsBean {
4
5     private String terms;
6     private String privacy;
7
8     public String getTerms() {
9         return terms;
10    }
11    public void setTerms(String terms) {
12        this.terms = terms;
13    }
14
15    public String getPrivacy() {
16        return privacy;
17    }
18    public void setPrivacy(String privacy) {
19        this.privacy = privacy;
20    }
21 }
22

```

< TermsBean.java >

### 3-2. Dao 패키지 생성

: 데이터베이스의 data에 접근하기 위한 객체 생성.



```

1 package kr.co.jboard1.dao;
2
3 import java.sql.Connection;
4
5 public class ArticleDao {
6
7     private static ArticleDao instance = new ArticleDao();
8     private ArticleDao() {}
9
10    public static ArticleDao getInstance() {
11        return instance;
12    }
13
14    public int[] getPageGroup(int currentPage, int lastPageNum) {
15
16        int groupCurrent = (int) Math.ceil(currentPage / 10.0);
17        int groupStart = (groupCurrent - 1) * 10 + 1;
18        int groupEnd = groupCurrent * 10;
19
20        if(groupEnd > lastPageNum) {
21            groupEnd = lastPageNum;
22        }
23
24        int[] groups = {groupStart, groupEnd};
25
26        return groups;
27    }
28
29    public int getPageStartNum(int total, int start) {
30        return total - start;
31    }
32
33    public int getLimitStart(int currentPage) {
34        return (currentPage - 1) * 10;
35    }
36
37    public int getCurrentPage(String pg) {
38
39        int currentPage = 1;
40
41        if(pg != null) {
42            currentPage = Integer.parseInt(pg);
43        }
44
45        return currentPage;
46    }
47
48    int lastPageNum = 0;
49
50    if(total % 10 == 0){
51        lastPageNum = total / 10;
52    }else{
53        lastPageNum = total / 10 + 1;
54    }
55
56    return lastPageNum;
57 }
58
59 public int selectCountArticle() {
60
61     int total = 0;
62
63     try{
64
65         Connection conn = DBConfig.getInstance().getConnection();
66         PreparedStatement psmt = conn.prepareStatement(Sql.SELECT_COUNT_ARTICLE);
67         ResultSet rs = psmt.executeQuery();
68
69         if(rs.next()) {
70             total = rs.getInt(1);
71         }
72
73         conn.close();
74     }catch(Exception e){
75         e.printStackTrace();
76     }
77
78     return total;
79 }
80
81 public void insertArticle() {}
82 public void insertComment(ArticleBean comment) {
83     try{
84
85         Connection conn = DBConfig.getInstance().getConnection();
86
87         PreparedStatement psmt = conn.prepareStatement(Sql.INSERT_COMMENT);
88         psmt.setInt(1, comment.getParent());
89         psmt.setString(2, comment.getContent());
90         psmt.setString(3, comment.getUId());
91         psmt.setString(4, comment.getRegIp());

```

이하 생략

### 3-3. db패키지 생성

: db 커넥션과 sql 구문 생성

```
1 package kr.co.jboard1.db;
2
3 import java.sql.Connection;
4
5
6 public class DBConfig {
7
8     private static DBConfig instance = new DBConfig();
9     private DBConfig() {}
10
11     public static DBConfig getInstance() {
12         return instance;
13     }
14
15     private final String HOST = "jdbc:mysql://54.180.160.240:3306/egag8843";
16     private final String USER = "egag8843";
17     private final String PASS = "1234";
18
19     public Connection getConnection() throws Exception {
20         Class.forName("com.mysql.jdbc.Driver");
21         Connection conn = DriverManager.getConnection(HOST, USER, PASS);
22
23         return conn;
24     }
25 }
26
```

< DBConfig.java >

```
1 package kr.co.jboard1.db;
2
3 public class Sql {
4
5     public static final String SELECT_COUNT_EMAIL = "SELECT COUNT(`email`) FROM `JBOARD_MEMBER` WHERE `email`=?";
6     public static final String SELECT_COUNT_HP = "SELECT COUNT(`hp`) FROM `JBOARD_MEMBER` WHERE `hp`=?";
7     public static final String SELECT_COUNT_NICK = "SELECT COUNT(`nick`) FROM `JBOARD_MEMBER` WHERE `nick`=?";
8     public static final String SELECT_COUNT_UID = "SELECT COUNT(`uid`) FROM `JBOARD_MEMBER` WHERE `uid`=?";
9
10     public static final String SELECT_MEMBER = "SELECT * FROM `JBOARD_MEMBER` WHERE `uid`=? AND `pass`=?";
11     public static final String INSERT_MEMBER = "INSERT INTO `JBOARD_MEMBER` SET "
12         + "`uid`=?, "
13         + "`pass`=PASSWORD(?), "
14         + "`name`=?, "
15         + "`nick`=?, "
16         + "`email`=?, "
17         + "`hp`=?, "
18         + "`zip`=?, "
19         + "`addr1`=?, "
20         + "`addr2`=?, "
21         + "`regip`=?, "
22         + "`rdate`=NOW()";
23
24     public static final String SELECT_TERMS = "SELECT * FROM `JBOARD_TERMS`";
25
26
27
28
29     public static final String SELECT_MAX_SEQ = "SELECT MAX(`seq`) FROM `JBOARD_ARTICLE`";
30     public static final String SELECT_COUNT_ARTICLE = "SELECT COUNT(*) FROM `JBOARD_ARTICLE` WHERE `parent`=?";
31
32     public static final String SELECT_ARTICLE = "SELECT * FROM `JBOARD_ARTICLE` AS a "
33         + "LEFT JOIN `JBOARD_FILE` AS b "
34         + "ON a.seq = b.parent "
35         + "WHERE a.`seq`=?";
36
37     public static final String SELECT_ARTICLES = "SELECT a.*, b.`nick` FROM `JBOARD_ARTICLE` AS a "
38         + "JOIN `JBOARD_MEMBER` AS b "
39         + "ON a.uid = b.uid "
40         + "WHERE `parent`=0 "
41         + "ORDER BY `seq` DESC "
42         + "LIMIT ?, 10";
43
44     public static final String SELECT_COMMENTS = "SELECT a.*, b.`nick` FROM `JBOARD_ARTICLE` AS a "
45         + "JOIN `JBOARD_MEMBER` AS b "
46         + "ON a.uid = b.uid "
47         + "WHERE `parent`=? "
48         + "ORDER BY `seq` ASC";
49
50
```

< Sql.java >