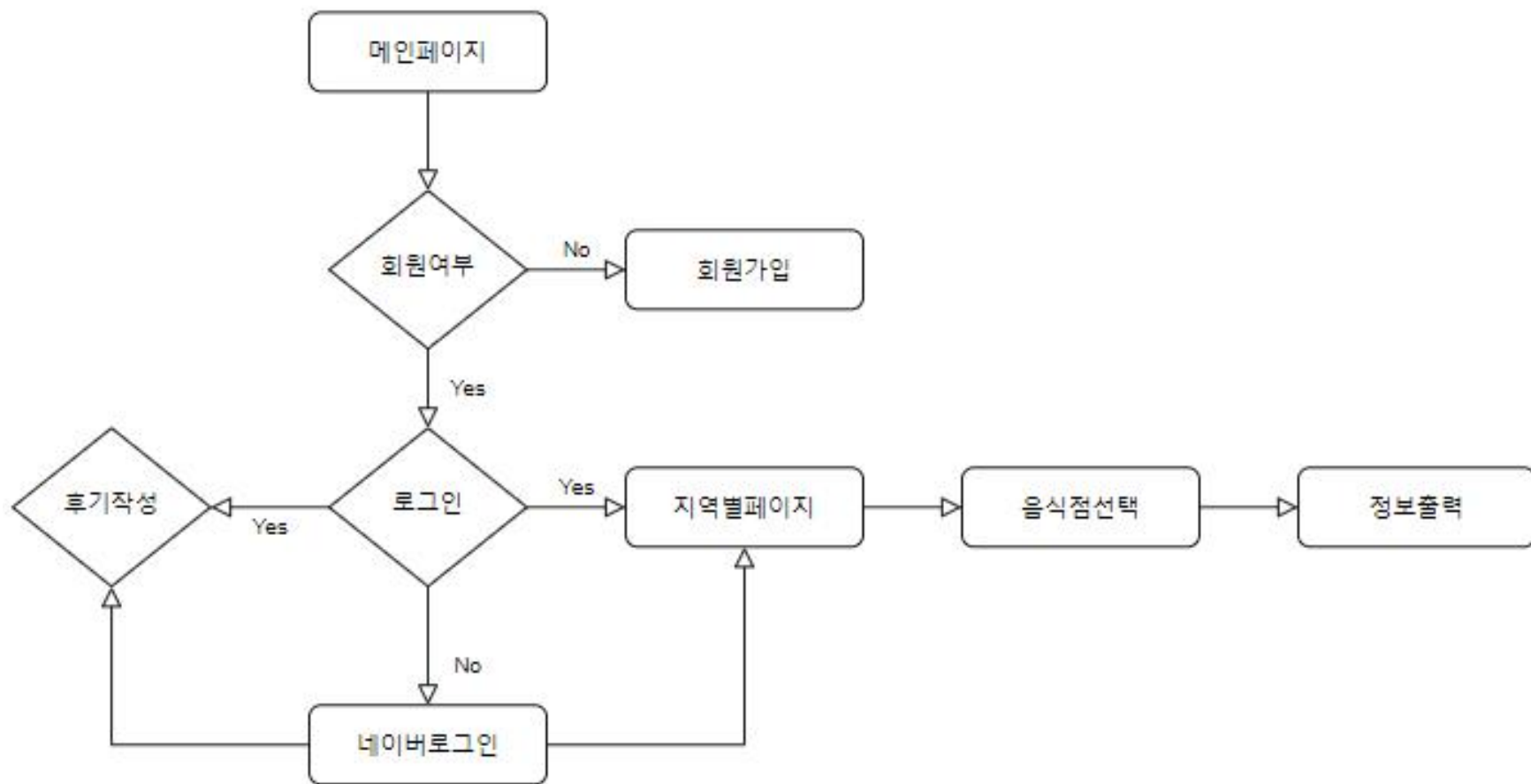


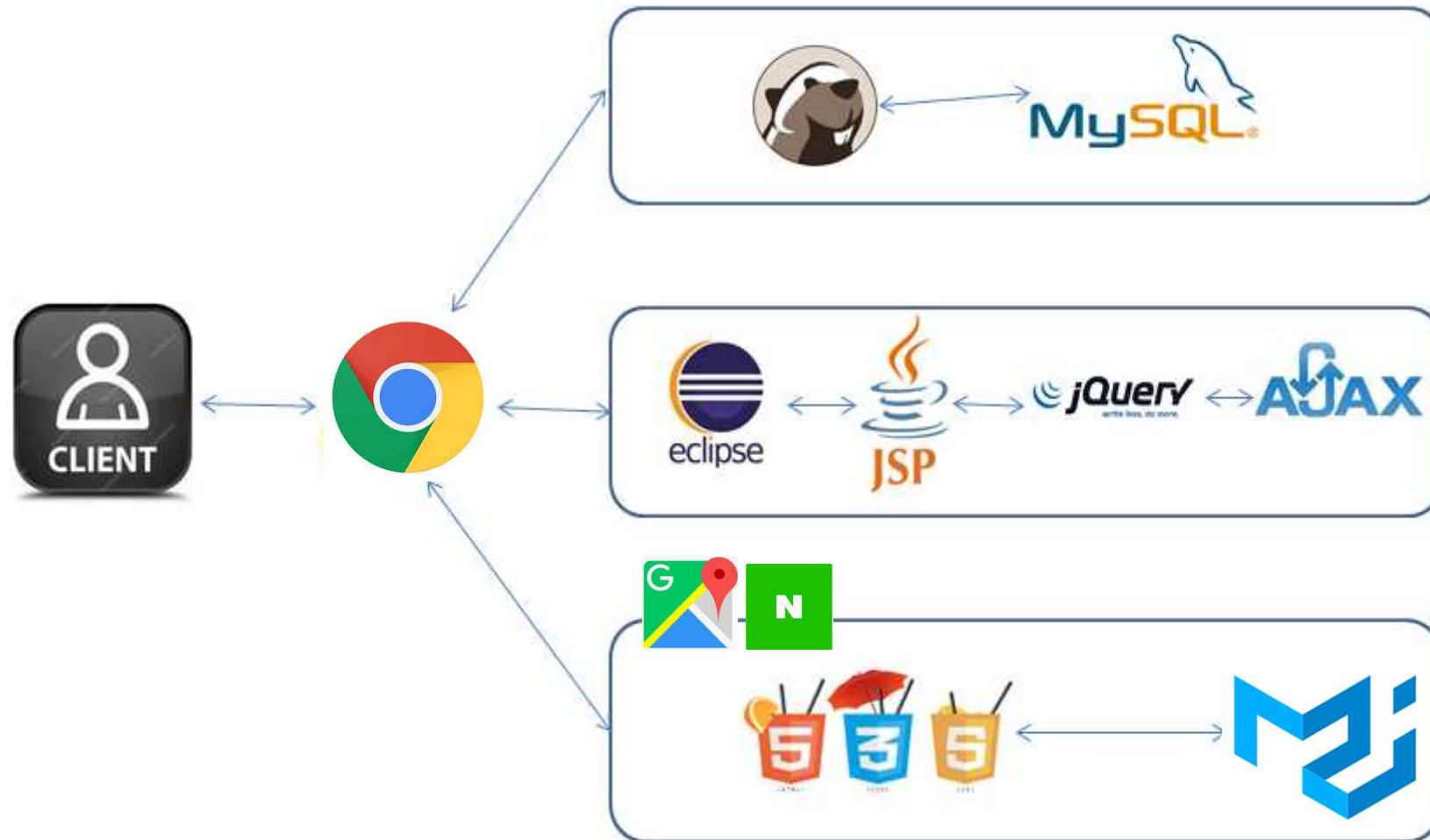
개요

프로젝트명	JAVA JSP MVC1-프로젝트 웹 어플리케이션 '어디서뭐먹지?'
개발기간	2019.09.16 ~ 2019.06.27
참여인원	3인
구현	메인지역별페이지, 음식점별상세조회, 회원기능, 후기작성
개발환경	Window10, Ubuntu 18.04
사용 도구	Eclipse, MySQL, Dbeaver, Github
사용 기술	JAVA, JSP, MySQL, Github, HTML, CSS, JS, JQUERY, Ajax, JSON
사용 외부 API	Kakao map API

요약흐름도



시스템구성도



1)메인(지역별조회)

필요구현	<p>특정 지역구를 선택 시, 해당지역의 음식점 조회.</p> 	<p>코드순서</p> <ol style="list-style-type: none"> 1) html내 해당 지역 클릭시, 처리될url+ db컬럼 값을 더해 Jsp 페이지로 전달한다. 2)연결된 jsp페이지에서 해당 컬럼의 값을 받은 후. 3) DAO의 selectAll메소드를 통해 ArrayList형식의 결과값을 받아낸 후, 4) jsp내 for문의 이용하여, <ul style="list-style-type: none"> - html에 정보를 담아낼 카드형식의 컴포넌트 - ArrayList의 인덱스 별 row로 담기는 값을 컬럼별로 해당 값을 필요한 자리에 삽입.
구현완료	<p>해당 지역의 음식점만 선택하여 나열.</p> 	<p>1) 연결처리할 jsp파일의 url과 해당지역의 값</p> <pre> <div class="box"> 마포구 </div> <div class="box"> 서대문구 </div> </pre>

1)메인(지역별조회)

- 2) 연결된 jsp페이지에서 해당 컬럼의 값과 DAO의 인스턴스로 메소드를 호출.

```
<%  
String district = request.getParameter("districtId");  
rstReadDAO dao=new rstReadDAO();  
ArrayList<rstDTO> list=dao.selectDistrict(district);
```

- 3) DAO의 selectAll메소드를 통해 ArrayList형식의 결과값


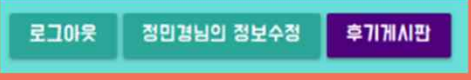
```
public ArrayList<rstDTO> selectDistrict(String district) throws Exception {  
    DBConnectionMgr mgr = DBConnectionMgr.getInstance();  
    Connection con = mgr.getConnection();  
    System.out.println("2. DB연결 ok.." + district);  
    String sql = "select * from rstdb where district = ?";  
    PreparedStatement ps = con.prepareStatement(sql);
```

```
    ps.setString(1, district);  
    System.out.println("3. SQL문 객체화 ok...");  
    System.out.println(ps + "sql쿼리확인");  
    ResultSet rs = ps.executeQuery();  
    System.out.println("4. SQL문 실행 요청 ok...");  
    ArrayList<rstDTO>list=new ArrayList<rstDTO>();  
    while (rs.next()) {  
        rstDTO dto2 = new rstDTO();  
        dto2.setRstid(rs.getString(1));  
        dto2.setRstname(rs.getString(2));  
        dto2.setDistrict(rs.getString(3));  
        dto2.setWorkingtime(rs.getString(4));  
        dto2.setTel(rs.getString(5));  
        dto2.setAddr(rs.getString(6));  
        dto2.setDesc(rs.getString(7));  
        dto2.setImg1(rs.getString(8));  
        dto2.setImg2(rs.getString(9));  
        dto2.setWriter(rs.getString(10));  
        list.add(dto2);  
    }  
    ps.close();
```

- 4) jsp의for문:
- 카드형식의 컴포넌트
- ArrayList인덱스 내 컬럼값

```
<%  
for(int i=0; i < list.size(); i++){  
    rstDTO dto2 = list.get(i);  
    %>  
  
    <div class="card"  
        style="width: 335px; height: 450px; margin-left: 50px;  
        margin-right: 25px;margin-bottom: 50px;">  
        <a href="rstDetails.jsp?rstId=<%=dto2.getRstid()%>">  
            <div class="boxpic" >  
                  
                <span class="card-title">  
                    <%=dto2.getRstname() %></span>  
            </a>  
        </div>  
        <div class="cardText">  
            <br>  
            <p><%=dto2.getDesc() %></p>  
        </div>  
    </a>  
    </div>  
    %>  
}
```

2)메인(헤드)

필요구현	 <p>로그인 후, 사용자 이름출력+ 컴포넌트 변경적용.</p>
구현완료	 <p>이름삽입 완료, 컴포넌트 변경적용 완료.</p>

코드순서

- 1) Html 로그인을 위한 팝오버 내 버튼에 폼 태그 적용.
- 2) 연결된 jsp페이지에서 id와 pw를 받은 후.
- 3) DAO의 check메소드를 통해
 - id와 pw를 DB와 대조하여 문제없는 경우: 리턴값1.
 - id는 존재하지만 pw가 맞지 않는 경우:리턴값 2.
 - id 가 존재하지 않는 경우 : 리턴값 3 으로 설정.
- 4) DAO의 리턴값을 받아
 - 리턴1의 경우: 세션항목과 값을 설정.
 - 리턴 2의 경우: "비번확인요망"의 경고창.
 - 리턴 3의 경우: "아이디확인요망"의 경고창 설정.
- 5) 세션값을 이어 받고자 하는 페이지에서get하여 연결 완성.

1) Html의 폼태그로 액션과 메소드를 정의한다.

```
<form action="session.jsp" method="POST">
  <div class="form-group" style="background-color:white; width:300px; height:225px; padding: 20px;
  position: fixed; left:74%; top:17%;">
    아이디<input type="text" name="id" placeholder="id" class="form-control"
    maxlength="16"> 비밀번호<input type="password"
    name="pw" placeholder="pw" class="form-control" maxlength="14">
    <button type="submit" id="myBtn" name="myBtn" class="btn btn-primary">Login >></button>
  </div>
</form>
```

2) id와pw를 변수에 담아 DAO의 check메소드에 적용.

```
<%
String id= request.getParameter("id");
String pw=request.getParameter("pw");
UserDAO dao=new UserDAO();
int result=dao.Check(id,pw);
```


2)메인(헤드)

3) DAO의 Check 메소드

```
public int Check(String id, String pw) throws Exception {
    Connection con = mgr.getConnection();
    String sql = "select pw from usermember where id=?";
    PreparedStatement ps = con.prepareStatement(sql);
    ps.setString(1, id);
    ResultSet rs = ps.executeQuery();
    int result = 0;
    if (rs.next()) {
        if (rs.getString("pw").equals(pw)) {
            result = 1; // 로그인성공
        } else {
            result = 2; // 비밀번호일치
        }
    } else {
        result = 3; // 아이디존재하지않음
    }
    ps.close();
    con.close();
    return result;
}
```

4) check메소드 리턴값에 따른 jsp의 수행내용

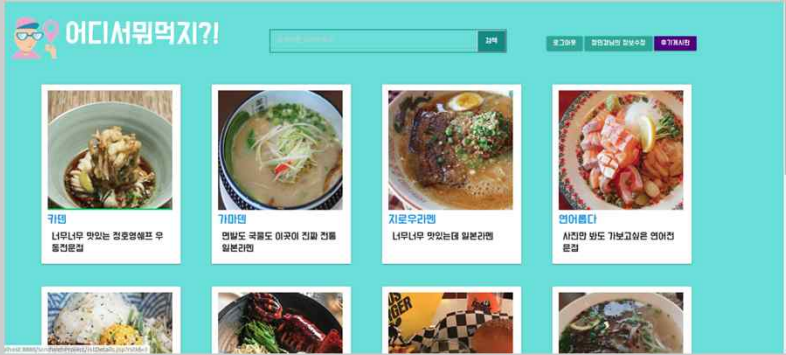
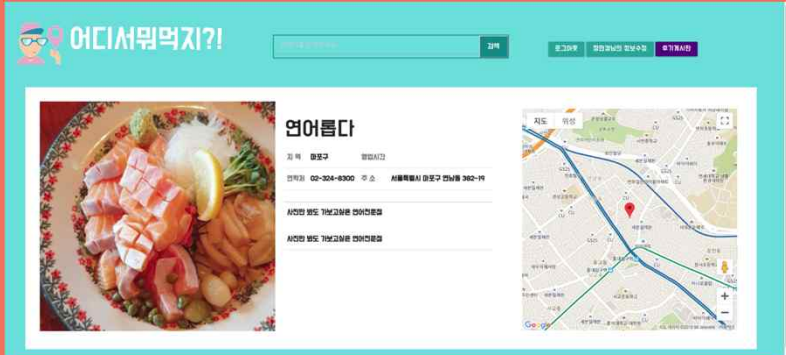
```
if (result==(1)){
    dto = new UserDTO();
    dto=dao.getUserInfo(id); //로그인회원정보담아오기
    session.setAttribute("id", id); //아이디세션설정
    session.setAttribute("pw", dto.getPw()); //아이디세션설정
    session.setAttribute("user_name", dto.getName()); //이름세션생성
    session.setAttribute("user_email", dto.getEmail()); //이메일세션생성
    session.setAttribute("user_phone", dto.getPhone()); //전화번호세션설정
    response.sendRedirect("index.jsp"); //다음페이지 설정

} else if (result==2) {
    %>
    <script type="text/javascript">
        alert("비밀번호를 확인해주세요.")
        location.href="index.jsp";
    </script>
    <%
} else if (result==3) {
    %>
    <script type="text/javascript">
        alert("아이디를 확인해주세요.")
        location.href="index.jsp";
    </script>
```

5) 로그인 후 연결 페이지에 세션get 처리.단, 세션의 값은 오브젝트 형식으로 넘어오므로, 형변환을 통해 사용하기 쉽게 스트링타입으로 별도 변수에 담준다.

```
String user_id= (String) session.getAttribute("id");
String user_name = (String) session.getAttribute("user_name");
%>
```

3)음식점조회/위치

필요구현	<p>특정 음식점을 클릭했을 때, 음식점정보와 지도검색.</p> 
구현완료	<p>특정 음식점의 정보와 위치가 검색된다.</p> 

코드순서

- 1)클릭시, 넘어갈url+ db컬럼 값을 더해 다음 Jsp로 전달
- 2)연결된 jsp페이지에서 해당 컬럼의 값을 받은 후.
DAO의 selectRst메소드를 통해 해당 값의 row결과를
- 3) 넘길 다음페이지 구성을 위한,
 - html에 정보를 담아낼 위치에 맞게 지정값 삽입,
 - 우측 구글맵api를 통해 해당 주소의 지도/마커삽입.

- 1) 넘어갈url+ db컬럼 값을 더해 다음 Jsp로 전달

```
<%
    int rstId = Integer.parseInt(request.getParameter("rstId"));
    rstReadDAO dao = new rstReadDAO();
    rstDTO dto = dao.selectRst(rstId);
```


3)음식점조회/위치

2) jsp페이지에서 해당 컬럼의 값 받아
DAO.selectRst메소드/
row결과

```
public rstDTO selectRst(int rstId) throws Exception {  
    DBConnectionMgr mgr = DBConnectionMgr.getInstance();  
    Connection con = mgr.getConnection();  
    System.out.println("2. db연결 ok!!!");  
    String sql = "select * from rstdb where rstid =?";  
    PreparedStatement ps = con.prepareStatement(sql);  
    ps.setInt(1,rstId);  
    System.out.println("3. sql문 객체화 ok!!!");  
    ResultSet rs = ps.executeQuery();  
    System.out.println("4. sql문 실행 요청 ok!!!");  
}
```


3) 넘길 다음페이지 구성을 위한 -
html에 정보를 담아낼 위치에 맞게
지정값 삽입

```
<div class="whiteboard">  
    <div class="sector" style="padding: 0;">  
          
    </div>  
    <div class="sector" style="margin: 0; margin-top: 2%;">  
        <h3><%=dto.getRstname()%></h3>  
        <table>  
            <tr>  
                <th style="color: gray;">지 역</th>  
                <th><%=dto.getDistrict() %></th>  
                <th style="color: gray;">영업시간</th>  
                <th><%=dto.getWorkingtime() %></th>  
            </tr>  
            <tr>  
                <th style="color: gray;">연락처</th>  
                <th><%=dto.getTel() %></th>  
                <th style="color: gray;">주 소</th>  
                <th><%=dto.getAddr() %></th>  
            </tr>  
            <tr>  
                <td colspan="4"><%=dto.getDesc() %> <br> <br> <br>  
                <%=dto.getDesc() %></td>  
            </tr>  
        </table>  
    </div>
```

우측 구글맵api를 통해 해당 주소의
지도와 마커 삽입.

```
<script> //상단 head 내 구글맵 스크립트와 API키를 준비,  
function initMap() {  
    var map = new google.maps.Map(document.getElementById('ggmap'), { //인스턴스생성  
        //맵을 생성할 div를 만들어 id를 'ggmap'으로, 해당구역에 맵을 출력.  
        zoom: 15,  
        center: {lat: -34.397, lng: 150.644} //맵의 디폴트값.  
    });  
    var geocoder = new google.maps.Geocoder();  
    //지오코더인스턴스.  
    geocodeAddress(geocoder, map); //어떤 위도&경도의 주소를 어느곳에 출력할지 함수호출의 파라미터값으로.  
    //geocodeAddress 함수호출, 파라미터 각각 넣어주고.  
}  
  
function geocodeAddress(geocoder, resultsMap) {  
    var address = "<%=dto.getAddr() %>"; //한글주소 변수  
    geocoder.geocode({'address': address}, function(results, status) { //콜백함수.  
        //한글주소받아서 적용하고, results는 데이터, status는 상태를 받아온다.  
        //아랫쪽 조건에 넣어서 지점을 지정하고, 마커를 더한다.  
        if (status === 'OK') {  
            resultsMap.setCenter(results[0].geometry.location);  
            var marker = new google.maps.Marker({  
                map: resultsMap,  
                position: results[0].geometry.location  
            });  
        } else {  
            alert('에러발생: ' + status);  
        }  
    });  
}  
initMap(); //함수 호출/
```

3) 키워드검색기능

필요구현	
구현완료	<p>검색창에 입력된 키워드에 따라 검색된 정보가 카드컴포넌트에 담겨 나온다.</p> 

코드
순서

- 1) 검색을 위한 버튼에 폼 태그 적용.
- 2) 연결된 jsp페이지에서 키워드 받은 후.
- 3) DAO의 search메소드를 통해 키워드가 해당하는 row의 값들을 ArrayList로 받는다.
- 4) jsp내 for문의 이용하여,
 - html에 정보를 담아낼 카드형식의 컴포넌트
 - ArrayList의 인덱스 별 row로 담기는 값을 컬럼별로 해당 값을 필요한 자리에 삽입.

1) 버튼에 폼 태그를 통해jsp로 이동

```
<form action="rstListOfSearch.jsp" method="GET">
    <input type="text" class="searchTerm" name="key" id="key"
        placeholder="검색어를 입력하세요." style="padding-left: 15px;">
    <button type="submit" class="searchButton">검색</button>
</form>
```

3) 키워드검색기능

- 2) 연결된 jsp페이지에서 키워드를 변수에 담아 DAO의 검색메소드에 적용.

```
<%  
String keyword = request.getParameter("key");  
rstReadDAO dao = new rstReadDAO();  
ArrayList<rstDTO> list=dao.selectSearch(keyword);
```

- 3) 키워드가 해당하는 특정값을 쿼리를 통해 검색하여 ArrayList로 받는다.

```
public ArrayList<rstDTO> selectSearch(String keyword) throws Exception {  
  
    DBConnectionMgr mgr = DBConnectionMgr.getInstance();  
    Connection con = mgr.getConnection();  
    System.out.println("2. DB연결 ok.." + keyword);  
    String sql = "SELECT * FROM rstdb WHERE rstname LIKE concat('%',?,'%')";  
  
    PreparedStatement ps = con.prepareStatement(sql);  
    ps.setString(1, keyword);  
    System.out.println("3. SQL문 객체화 ok...");  
    System.out.println(ps + "sql쿼리확인");  
    ResultSet rs = ps.executeQuery();  
    System.out.println("4. SQL문 실행 요청 ok...");  
  
    ArrayList<rstDTO> list=new ArrayList<rstDTO>();
```

- 4) jsp내 for문의 이용하여,
- html에 정보를 담아낼 카드형식의 컴포넌트
- ArrayList의 인덱스 별 row로 담기는 값을 컬럼별로 해당 값을 필요한 자리에 삽입.

```
<%  
for(int i=0; i < list.size(); i++){  
    rstDTO dto2 = list.get(i);  
    %>  
    <div class="card"  
        style="width: 335px; height: 450px; margin-left: 50px; margin-right: 25px; margin-bottom: 50px;">  
        <a href="rstDetails.jsp?rstId=<%=dto2.getRstId()%>">  
            <div class="boxpic">  
                 <span class="card-title">  
                    <%=dto2.getRstname() %></span>  
            </a>  
        </div>  
        <div class="cardText">  
            <br>  
            <p><%=dto2.getDesc() %></p>  
        </div>  
        </a>  
    </div>  
    %>  
}
```