

Soldesk: 2022_R802

Team4 Semi Project



<https://github.com/LitaHart/SemiProject4Fes.git>

目次

1. チームメンバー紹介
2. テーマと選定理由
3. プロジェクト目標
4. 開発スケジュール
5. 一次会議結果
6. 作業ツール
7. ウェブサイト機能紹介
8. ウェブサイトDB紹介
9. コードレビュー(パート別)
10. 感想

1. チームメンバー紹介

				
氏名	田侑眞 (ジョン・ユジン)	朴旼智 (パク・ミンジ)	林東勲 (イム・ドンファン)	朴視願 (パク・シウォン)
e-mail	lita.y.jeon@gmail.com	busy8863@naver.com	dla_ehdgns@naver.com	soobin414@gmail.com
担当	チームリーダー デザイン/ CSS / 掲示板 / DB管理 / ウィジェットなど	CSS / 祭り情報パース/ (祭り情報一覧) / DB管理など	CSS / ホームのスライダーなど	地図API適用 など

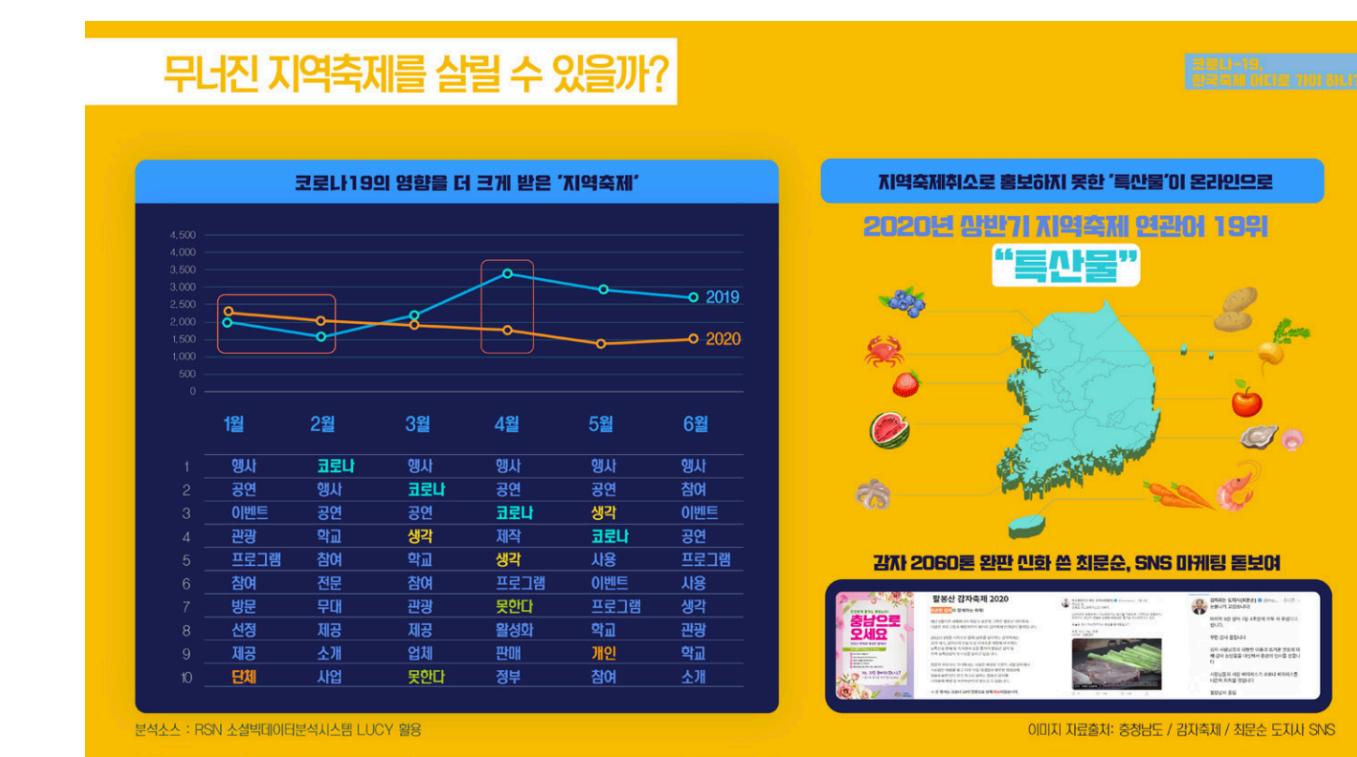
2. テーマと選定理由

テーマ：ソウル地域の祭りを紹介するウェブサイトを作る。

日本の文化に興味を持っている人たちが集まってチームを組みました

テーマ選定会議中、日本各地で行われるお祭りについて話し合うようになりました

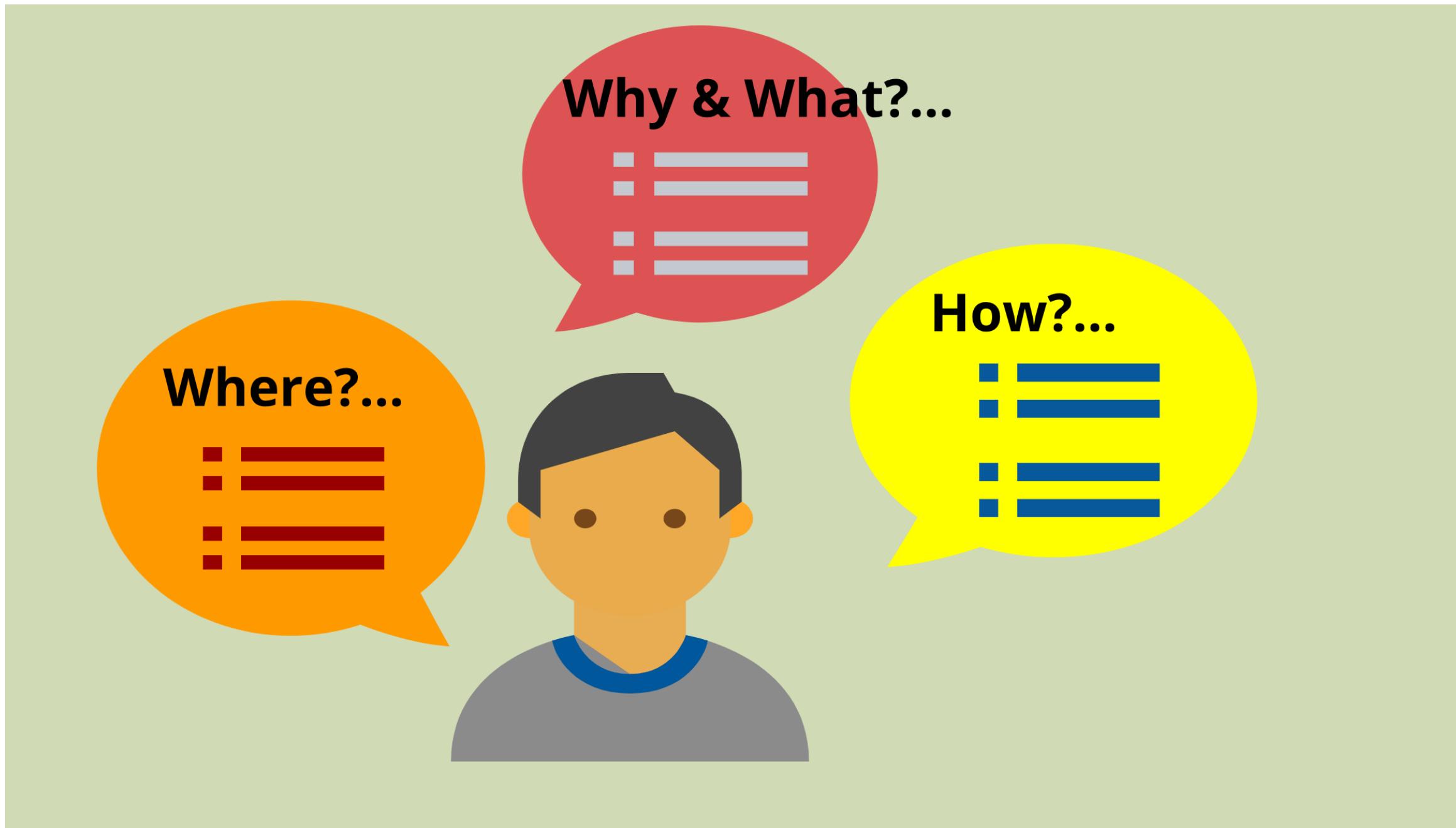
情報収集中、『新型コロナウイルスの影響により、旅行と祭りに対する大衆の関心が減少したが、地域経済の発達のためには地域が自主的に主管するお祭りが大きく役立つ』という韓国の統計結果から、韓国も地域のお祭りの情報を集めたウェブサイトの必要性を感じ、テーマとして決定しました。



イメージ：韓国観光公社祝祭オンラインフォラム資料集より

3. プロジェクト目標(Review & Challenge)

1. Java、JS、HTML・css、などのプログラミング言語を応用し、慣れるようになります。
2. 普段やってみたかった応用法・プラグインなどを適用してみること。
この二つを目標に進めました。



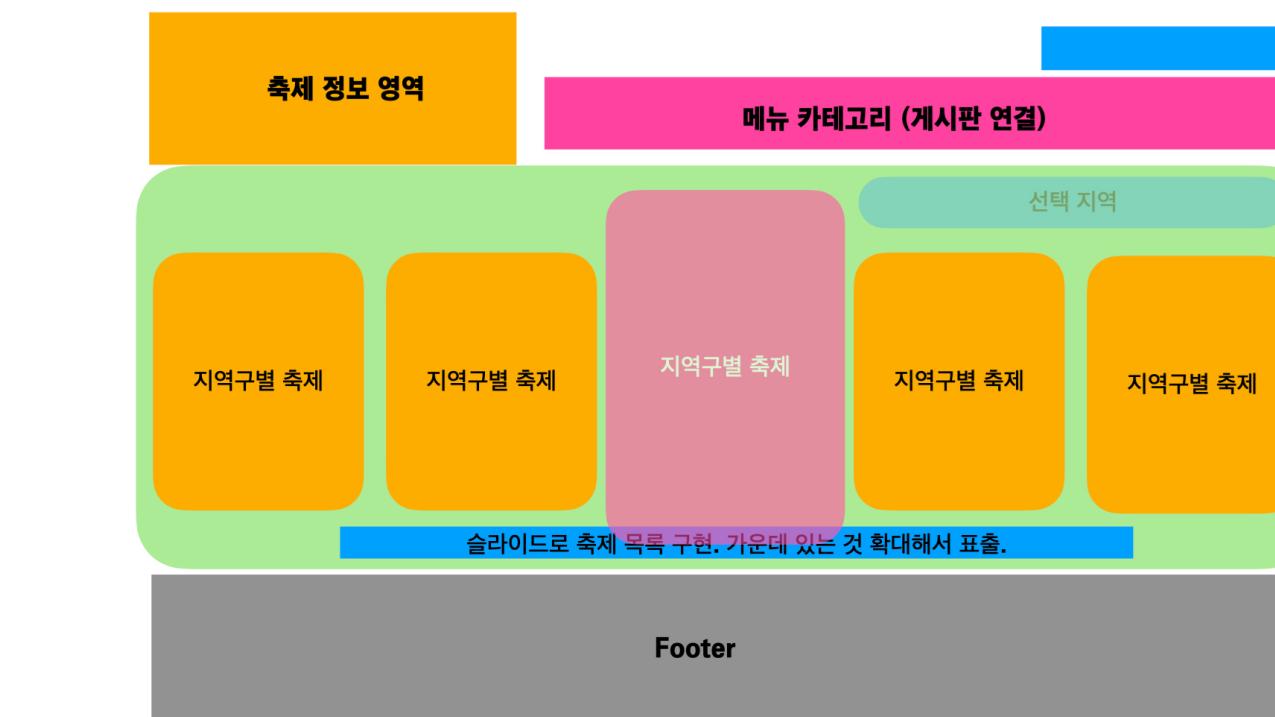
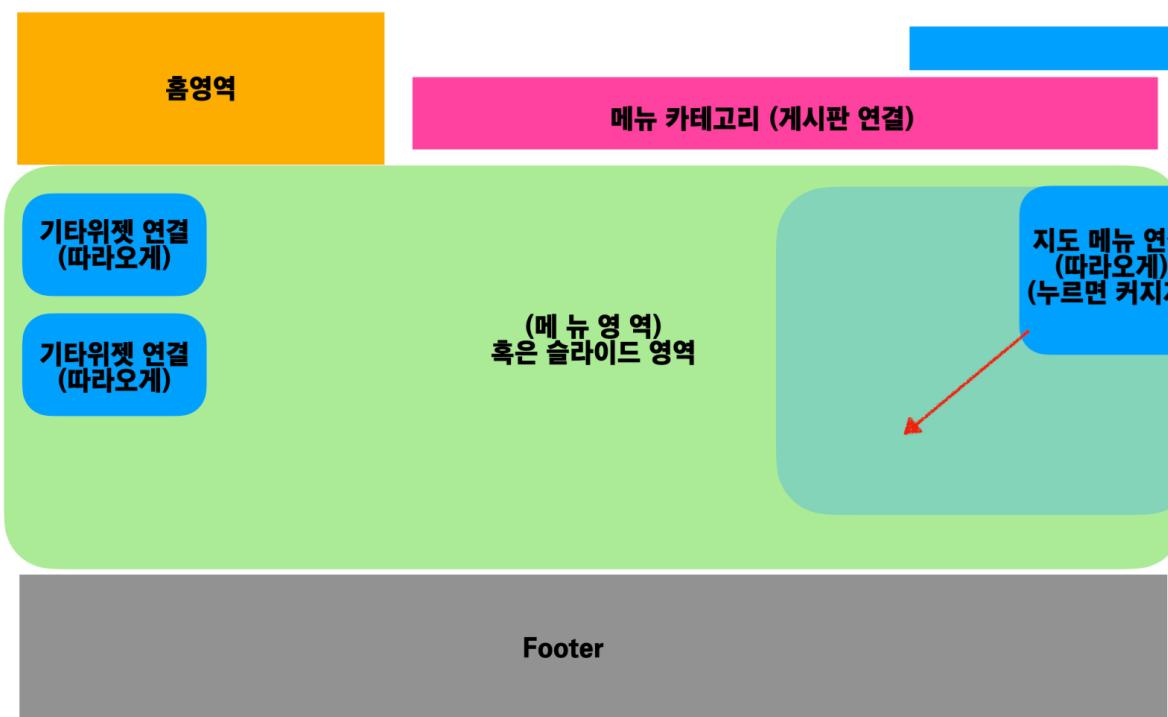
4. 開発スケジュール

Semi Project Team4. Schedule

	Mon	Tue	Wed	Thur	Fri	Sat (Weekend)
Week1	6.13	6.14	6.15	6.16	6.17	
	システム具現化	システム具現化	システム具現化	システム具現化	システム具現化	
	CSS					
	6.20	6.21	6.22	6.23	6.24	
	システム具現化	システム具現化	システム具現化	CSS	CSS	個人のスケジュール によって流動的に
Week2	6.27	6.28	6.29	6.30	7.1	
	CSS	エラーチェック及び デバッグ	エラーチェック及び デバッグ	プレゼンテーション 準備	プレゼンテーション 準備	
	システム具現化			エラーチェック及び デバッグ		
Week3						

5. 一次会議結果

1. メインテーマ決定
2. KakaoAPIなどオープンソースコードを用いて地図(道案内)具現
3. クリックに反応するスライダー具現(一般型、地図型)
4. 掲示板具現(会員登録機能なし)
5. ウェブサイト内のテキスト翻訳機能(プラグイン)
6. 参考資料・サイト探索(韓国観光公社・ソウルオープンデータ広場・カカオデベロッパーなど)
7. ホームページデザイン試案構想(下)



6. 作業ツール

連絡



開発環境



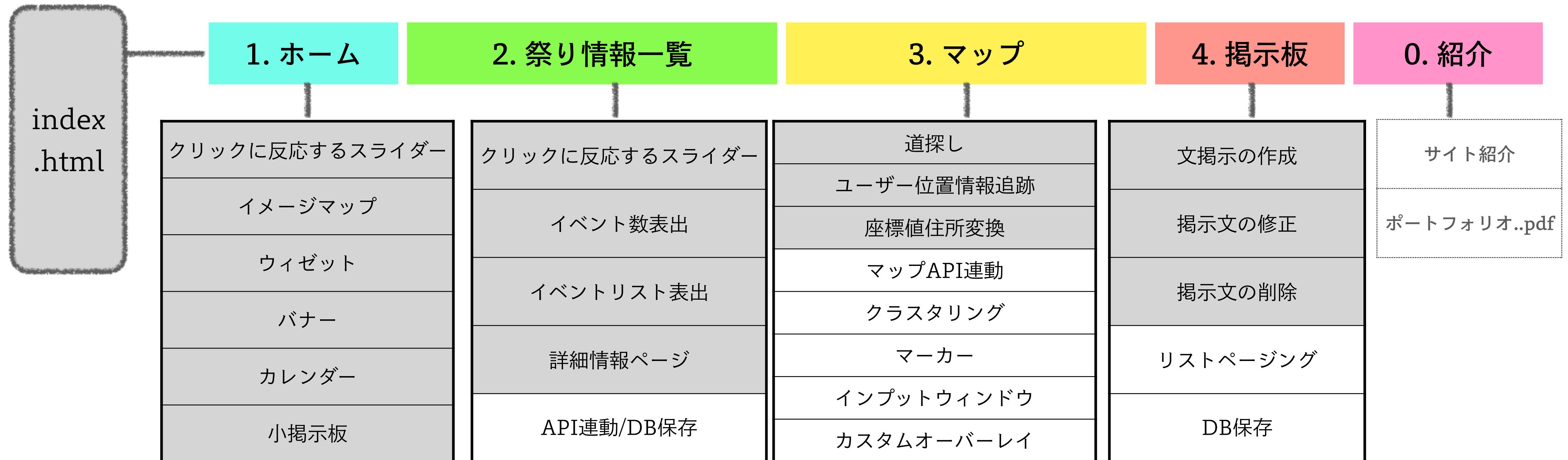
使用言語



jQuery
write less, do more.



7. ウェブサイト機能紹介



8. サイトDB紹介



1. ホーム

2. 祭り情報一覧

3. マップ

4. 揭示板

0. 紹介

日本語版DB

```
create table festival_list_jp(
    m_no number(5) primary key,
    codename varchar2 (500 char) not null,
    guname varchar2 (500 char) not null,
    title varchar2 (500 char) not null,
    fdate varchar2 (500 char) not null,
    place varchar2 (500 char) not null,
    program varchar2 (600 char),
    orglink varchar2 (500 char),
    mainimg varchar2 (500 char) not null,
    usetarget varchar2 (500 char) not null,
    usefee varchar2 (500 char)
);
```

韓国語版DB

```
create table festival_list(
    m_no number(5) primary key,
    codename varchar2 (500 char) not null,
    guname varchar2 (500 char) not null,
    title varchar2 (500 char) not null,
    fdate varchar2 (500 char) not null,
    place varchar2 (500 char) not null,
    program varchar2 (600 char),
    orglink varchar2 (500 char),
    mainimg varchar2 (500 char) not null,
    usetarget varchar2 (500 char) not null,
    usefee varchar2 (500 char)
);
```

掲示板DB（日韓兼用）

```
CREATE TABLE BOARD_DB (
    B_NO NUMBER(3) PRIMARY KEY,
    B_CATE VARCHAR2(50 CHAR) NOT NULL,
    B_TITLE VARCHAR2(200 CHAR) NOT NULL,
    B_NAME VARCHAR2(30 CHAR) NOT NULL,
    B_TXT VARCHAR2(1000 CHAR) NOT NULL,
    B_IMG VARCHAR2 (200CHAR),
    B_DATE DATE NOT NULL,
    B_PW VARCHAR2(20CHAR) NOT NULL
);

CREATE SEQUENCE BOARD_SEQ;
```

9. コードレビュー(パート別)

カテゴリ「掲示板」「サイト情報」（担当:ジョン・ユジン）

担当機能

1. プロジェクト総括及び会議進行
2. 全体ホームページCSS及びロゴデザイン、和訳プラグイン適用、ウィジェット適用など
3. カテゴリ「掲示板」ページ機能具現
 - ユーザーが入力したデータをDBに保存
 - 画像アップロード及び修正機能
 - ボードの削除/DBデータの削除機能
 - 掲示板リスト表出・ページング機能
4. カテゴリ「サイト情報」掲示のppt製作及び詳細ページ作成
5. ppt・サイト和訳。

使用機能・ツール

- eclipse(html, jsp, js, cssなど)
- Oracle SQL Developer
- Oracle Database Actions(SQL)

掲示板: ボードアップデート

ホーム 記事정보일람 지도보기 게시판 사이트정보

글 수정하기

제목 테스트 게시글
카테고리 자유게시판

닉네임 닉네임 테스트

이건 비밀이야
아무에게도 고백하지 않았던
이야기를 들려주면 큰 눈으로 너는 묻지
How, wow, wow, whatever
나 실은 말이야
저기 아득한 미래로부터 날아왔어
쏟아질 듯이 빼곡한 별들 사이를 지나
Fly, fly, fly
있지 그곳도 사실 바보들투성이야
아니 매우 반짝이는 건 오히려
Now, now, now
이 하루 이 지금 우리 눈부셔 아름다워
이 불꽃놀이는 끝나지 않을 거야
Oh, Whatever
흐린 날이면
거짓말처럼 무섭게 깜깜했지

사진첨부 Choose File No file chosen
갱신이 없을 시, 기존 이미지가 적용됩니다.

비밀번호 설정
작성완료

```
// 글 업데이트하기.  
public void updateBoard(HttpServletRequest request) {  
    PreparedStatement pstmt = null;  
    try {  
  
        String saveDirectory = request.getServletContext().getRealPath("fileFolder");  
        System.out.println(saveDirectory);  
  
        MultipartRequest mr = new MultipartRequest(request, saveDirectory, 31457280, "UTF-8",  
            new DefaultFileRenamePolicy());  
  
        if (mr.getFilesystemName("fName") != null) {  
            String sql = "UPDATE BOARD_DB SET b_title= ?, b_cate=?,b_name=?,b_txt=?,b_img=?,b_pw=? WHERE b_NO=?";  
            pstmt = con.prepareStatement(sql);  
  
            String title = mr.getParameter("title");  
            String cate = mr.getParameter("boardType");  
            String name = mr.getParameter("name");  
            String txt = mr.getParameter("txt");  
            txt=txt.replace("<br>","\r\n");  
            String img = mr.getFilesystemName("fName");  
            String pw = mr.getParameter("password");  
            String number = mr.getParameter("number");  
  
            // 값 받고, ?에 셋팅.  
            pstmt.setString(1, title);  
            pstmt.setString(2, cate);  
            pstmt.setString(3, name);  
            pstmt.setString(4, txt);  
            pstmt.setString(5, img);  
            pstmt.setString(6, pw);  
            pstmt.setString(7, number);  
  
        } else {  
            String sql = "UPDATE BOARD_DB SET b_title= ?, b_cate=?,b_name=?,b_txt=?,b_pw=? WHERE b_NO=?";  
            pstmt = con.prepareStatement(sql);  
  
            String title = mr.getParameter("title");  
            String cate = mr.getParameter("boardType");  
            String name = mr.getParameter("name");  
            String txt = mr.getParameter("txt");  
            txt=txt.replace("<br>","\r\n");  
            String pw = mr.getParameter("password");  
            String number = mr.getParameter("number");  
        }  
    } catch (Exception e) {  
        e.printStackTrace();  
    } finally {  
        if (pstmt != null) {  
            try {  
                pstmt.close();  
            } catch (SQLException e) {  
                e.printStackTrace();  
            }  
        }  
    }  
}
```

写真添付によってif文処理

1.ボタンを押すときパラメータで値を持ってくる。
2.DBに接続してSQL文に修正

1.改行処理

掲示板: ボードローディング



```
// 전체 게시판 글 불러오기  
public void getAllBoard(HttpServletRequest request) {  
    PreparedStatement pstmt = null;  
    ResultSet rs = null;
```

```
try {  
    String sql = "select*from BOARD_DB order by B_DATE desc";
```

```
pstmt = con.prepareStatement(sql);  
rs = pstmt.executeQuery();
```

```
// 가변배열에 넣기  
ArrayList<Board> boards = new ArrayList<Board>();
```

```
Board b = null;
```

```
while (rs.next()) {  
    b = new Board(rs.getInt("b_no"), rs.getString("b_cate"), rs.getString("b_title"),  
        rs.getString("b_name"), rs.getString("b_txt"), rs.getString("b_img"), rs.getDate("b_date"),  
        rs.getString("b_pw"));  
    boards.add(b);
```

```
}  
request.setAttribute("boards", boards);  
  
} catch (Exception e) {  
    // TODO: handle exception  
    e.printStackTrace();  
} finally {  
    DBManager_Main.getDbm().close(null, pstmt, rs);  
}
```

```
<div class="BoardSeeNoteDiv">
```

```
<!-- 글쓰기버튼! -->  
<div class="BoardInputDiv">  
    <button id="btn_write" onclick="location.href='BoardInputHC'">글쓰기</button>  
</div>
```

```
<div id="boardSeeNote">  
    <!-- 글 목록 출력 -->  
    <table id="seeNoteTable" border="1">  
        <tr id="seeNoteTopTr">  
            <td>게시판 분류</td>  
            <td>글 제목</td>  
            <td>작성자</td>  
            <td>작성일자</td>  
        </tr>
```

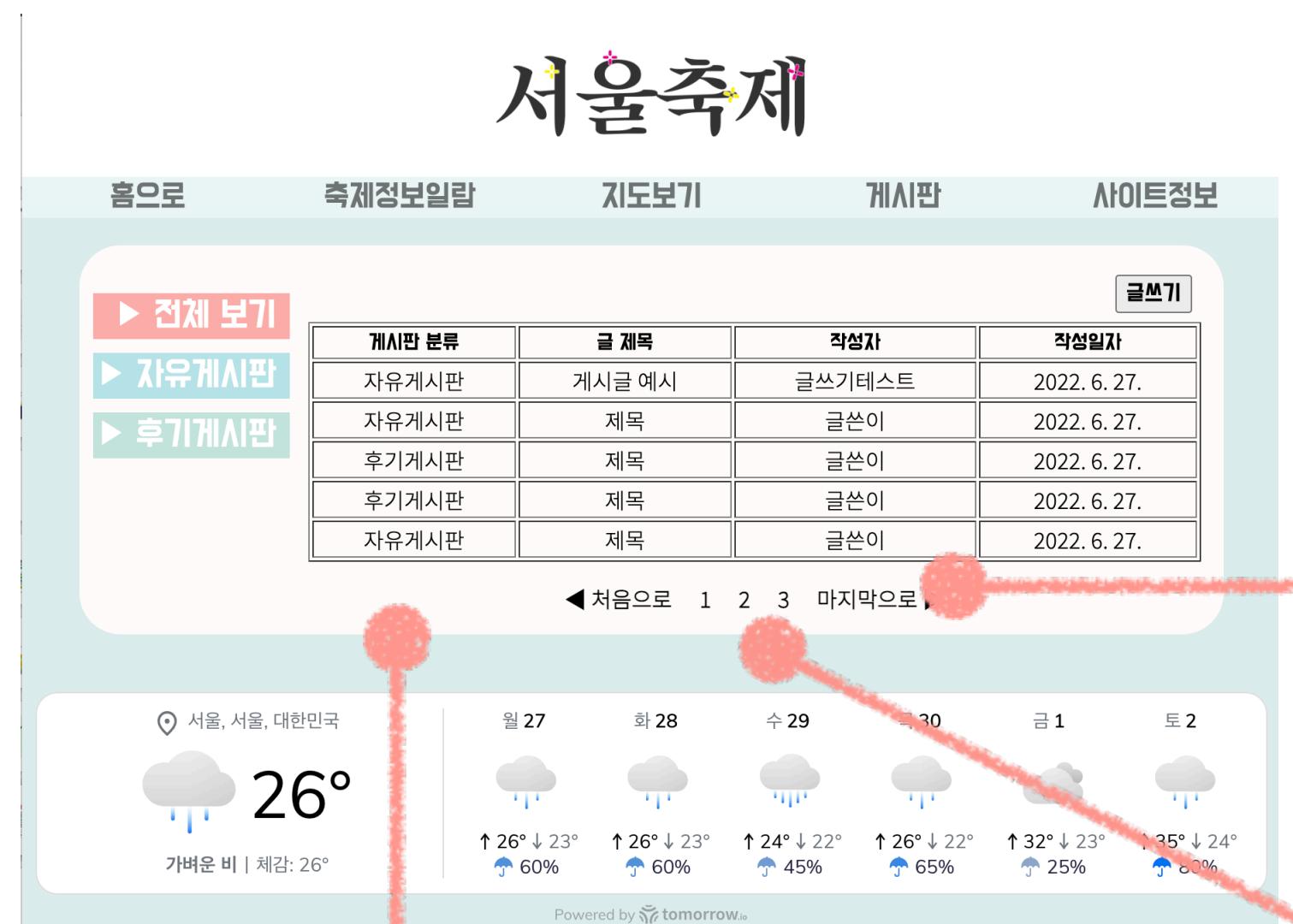
```
<!-- 배열에 넣은 것을 출력-->  
<c:forEach var="b" items="${boards}">  
    <tr id="seeNoteTable_detail">  
        <td>${b.cate }</td>  
        <td>${b.title}</td>  
        <td>${b.name}</td>  
        <td><fmt:formatDate value="${b.date}" /></td>  
    </tr>  
</c:forEach>  
</table>  
</div> <!-- boardSeeNote 닫음 -->
```

1.SQL文で全体掲示文の照会(最近の順に整列)

2.Boardオブジェクトを作ってAttributeに転送

EL文を使用して出力
JSTL反復文を使用

掲示板: ページング



```
select count(*) from BOARD_DB;
select * from (
    select rownum as rn, b_no, b_cate, b_title, b_name, b_txt, b_img, b_date, b_pw
    from (
        select * from BOARD_DB where B_CATE = '자유게시판'
        order by b_date desc
    ) where rn between 1 and 5 ;
```

1.SQL文を使って掲示文の順番を再整列(ROWNUM)

```
// 페이지 관련
// 마지막페이지 계산
public void lastPage(HttpServletRequest request) {
    PreparedStatement pstmt = null;
    ResultSet rs = null;

    String cate = request.getParameter("cate");
    String sql = "";

    // 글이 몇 개인지 찾기
    try {
        // if문 처리
        if (cate.equals("all")) {
            sql = "select count(*) from BOARD_DB";
            pstmt = con.prepareStatement(sql);
        } else if (cate.equals("free") || cate.equals("review"))
            sql = "select count(*) from BOARD_DB WHERE B_CATE IN
                if (cate.equals("free")) {
                    cate = "%자유게시판%";
                } else if (cate.equals("review")) {
                    cate = "%후기게시판%";
                }
            pstmt = con.prepareStatement(sql);
            pstmt.setString(1, cate);
        }
        // SQL문 실행하기
        rs = pstmt.executeQuery();

        if (rs.next()) {
            int total = rs.getInt("count(*)");
            int lastPage = (int) Math.ceil(total/5);
            request.setAttribute("lastpage", lastPage);
        }
    } catch (Exception e) {
        e.printStackTrace();
    } finally {
        DBManager_Main.getDbm().close(null, pstmt, rs);
    }
}
```

```
<!-- 페이지 관련 -->
<div class="pagingDivMain">
    <div id="pagingTriDiv" onclick="location.href='BoardViewHC?vPage=1&lastPage=${lastpage}&cate=${param.cate}'><a href="#">처음으로</a></div>
    <br>
    <c:forEach var="i" begin="1" end="${lastpage}">
        <div id="pagingNumDiv" onclick="location.href='BoardViewHC?vPage=${i}&lastPage=${lastpage}&cate=${param.cate}'>
            <a href="#"><${i}></a>
        </div>
    </c:forEach>
    <div id="pagingTriDiv" onclick="location.href='BoardViewHC?vPage=${lastpage}&lastPage=${lastpage}&cate=${param.cate}'><a href="#">마지막으로</a></div>
</div>
```

1.EL文とパラメータでユーザーが選択したページを表示する

1. SQL文で全体ページ数を確認

2. 数式使用してページに表示される掲示板ローディング

カテゴリ 「祭り情報一覧」（担当: パク・ミンジ）

担当機能

- ソウル公共データ広場(<https://data.seoul.go.kr/>)でソウル市の文化・イベント情報APIをパース
- カテゴリー「祭り情報一覧」ページの機能を具現
 1. ソウル市の「区」を選択した時、パースしたデータをスライドに表示
 2. 「区」を選択した場合、各区で開催されるイベント(お祭り)の数をローディング
 3. 「もっと見る」をクリックすると、各区で開催されるイベントリスト表出、イベントページに移動。
 4. 各イベントごとに詳細情報ページ具現
- メインホームページの中間部分CSS担当及び和訳担当
 1. 「祭り情報一覧ページへ」ショートカットバナー製作
 2. 揭示板の最近の掲示文5つまでメインホームページに露出されるように具現
 3. カレンダーを挿入してさらに表示を押した時、カレンダーとWeekListに祭りのイベント期間が表出されるように具現
 - + お祭りタイトルをクリックした時、詳細情報ページへ移動

使用機能・ツール

- eclipse(html, jsp, js, ajax, css …)
- Oracle SQL Developer
- Oracle DB

festival_list table

The screenshot shows the Seoul Festival website. At the top, there's a navigation bar with tabs like 'Home', 'Festival Information' (which is highlighted with a red box), 'Map', 'Post', and 'Site Information'. Below the navigation, there's a large image of a purple festival float. To the right of the image, there's a box containing festival details: 'Festival Information' (25 items), 'Cover Image', 'Title', and 'Main Image'. At the bottom, there's a weather forecast for Seoul.

1

```

function setImgFromAddr() {
    var langSelect = document.getElementById("selectbox");

    // select element에서 선택된 option의 value가 저장된다.
    var selectValue = langSelect.options[langSelect.selectedIndex].value;

    // select element에서 선택된 option의 text가 저장된다.
    var selectText = langSelect.options[langSelect.selectedIndex].text;
    selectText = encodeURI(selectText);

    // ajax 통신
    $.ajax({
        type : "GET", // HTTP method type(GET, POST) 형식이다
        dataType : "text",
        url : "mj_fesInfo_board/imgFromAddr.jsp", // 컨트롤러에서 대기중인 URL 주소이다.
        data : {
            guName : selectText
        }, // Json 형식의 데이터이다.
        success : function(data) { // 비동기통신의 성공일경우 success콜백으로 들어옵니다. 're
            const obj = JSON.parse(data);
            let stackpk = $(".photostack-pk");
            let stackimg = $(".photostack-img").children();
            let stacktitle = $(".photostack-title");

            // ----- 더미
            let stackimg2 = $(".coverImg").children();
            let stacktitle2 = $(".coverTitle");
            let objLength = obj.result.length;
            console.log(objLength);
            $("#numberOfFes").text(objLength); ←各区の祭りの個数をカウント

            let pk;
            let stitle;
            let mimg;
            let j = 0;

            for (var i = 0; i < 16; i++) {
                j = i % objLength;
                ppk = obj.result[j].m_no;
                stitle = obj.result[j].title;
                mimg = obj.result[j].mainImg;

                let pkpk = stackpk[i];
                let m_no = "festivalInfoCon?m_no=" + ppk;

                $(pkpk).attr("href", m_no);
                let simg = stackimg[i];
                $(simg).attr("src", mimg);
                let settile = stacktitle[i];
                $(settile).text(stitle);
            }
        }
    });
}

```

2

```

PreparedStatement pstmt = null;
ResultSet rs = null;

try {
    String sql = "select * from festival_list where guName=?";
    Connection con = DBManager_Main.getDbm().connect();
    pstmt = con.prepareStatement(sql);
    String guName = (String) request.getParameter("guName");
    guName = URLDecoder.decode(guName, "UTF-8");

    pstmt.setString(1, guName);
    rs = pstmt.executeQuery();

    ArrayList<festival> festivals = new ArrayList<festival>();
    festival f = null;
    JSONArray arr = new JSONArray();

    int i = 0;
    while (rs.next()) {
        JSONObject obj = new JSONObject(); // {}
        obj.put("m_no", rs.getString("m_no"));
        obj.put("title", rs.getString("title"));
        obj.put("mainImg", rs.getString("mainImg"));
        arr.add(i, obj);
        i++;
    }

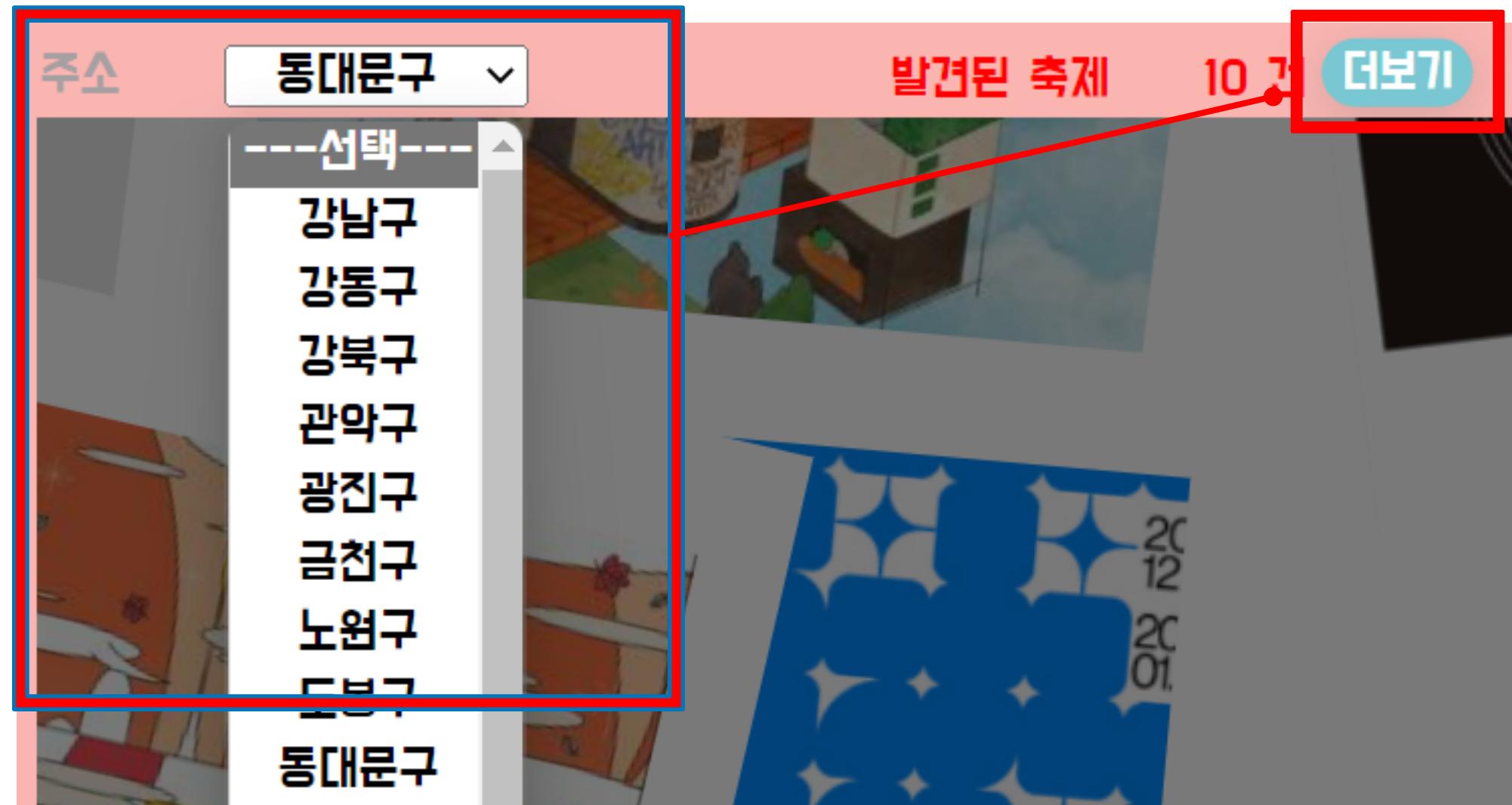
    // json 형태로 리턴하기 위한 json 객체 생성
    JSONObject jobj = new JSONObject(); // {}
    jobj.put("result", arr);

    // 응답시 json 타입이라는 걸 명시 ( 안해주면 json 타입으로 인식하지 못함 )
    response.setCharacterEncoding("UTF-8");
    response.setContentType("application/json");
    out.print(jobj.toString()); // json 형식으로 출력

} catch (Exception e) {
    System.out.println(e);
}

finally {
    DBManager_Main.getDbm().close(null, pstmt, rs);
}

```



1

```
<script type="text/javascript">
$(document).ready(function() {
    $("#fesList").on("click", function() {
        //select라는 태그 이름을 이용해서 select 박스를 선택한 후
        // .val() 함수로 값을 가져오면 option의 value 값을 가져온다.
        //따라서 val1, val2, val3...의 값을 가져온다.

        if ($("#select option:selected").val() == "") {
            alert('구를 선택하지 않으셨습니다.');
            location.href = "festivalHC";
        } else {
            var kkk = $("#select option:selected").val();
            alert(kkk);
            location.href = "festivalListCon?Seoul=" + kkk;
        }
    });
});
```

↑ボタンを押したとき選択された区のリストページに移動

←Exception

[서울문화재단] 2022 거리예술 시즌 프로그램 (거리예술 카라반 '봄')	2021 서울라이트	빅무브 with Lia KIM	[DDP] <집합 이론> 展	<2021 DDP 디자인 마켓 판매자 모집>	[서울문화재단] 2021 서울국제작가축제
Winter Seoul 2021 : 기후변화 메가시티 그리고 봄	2021 거리예술 시즌 프로그램(카라반)	〈 단풍 빛고 소풍 가자! 〉 홍릉단풍길 스텁프투어	2022 선농대제 개최		

2

```
<div id="seoulGu">${param.Seoul}축제</div>
<div id="festivalListDiv" style="display:inline-table;">
    <c:forEach items="${fes}" var="f">
        <table id="festivalListTbl" border="1" width="170px" height="250px;" style="float: left; border: 1px solid gray">
            <tr>
                <td><a href="festivalInfoCon?m_no=${f.no}"></a></td>
            </tr>
            <tr style="height: 90px;">
                <td class="festivalListTitle"><a href="festivalInfoCon?m_no=${f.no}">${f.title}</a></td>
            </tr>
        </table>
    </c:forEach>
</div>
```

クリックの時、そのイベントの情報を見せてあげる



祭りバナーの上にマウスを乗せたら
写真の後に影の領域が減りながら輝く効果が現れ、
「Click me!」という文字が現れる。



```
<!-- 축제정보 보러가기 영역 -->
<div class="photo">
  <span id="FestivalTxt">축제정보<br> 보러가기</span>
  <a href='festivalHC'>
    <div class="glow-wrap" >
      <i class="glow"></i> ←傾き効果
    </div></a>
  <p id="click">→ Click me!</p>
</div>
```

1 影エフェクト

```
.photo img {
  object-fit: cover;
  box-shadow: 10px 15px 25px 0 rgba(0, 0, 0, .2);
  display: block;
  transition: all .5s cubic-bezier(0.645, 0.045, 0.355, 1);
}
```

Hoverすると影の領域が減る。

```
.photo:hover img {
  box-shadow: 1px 1px 10px 0 rgba(0, 0, 0, .1);
}
```

2 ‘Click me!’ 表示

```
#click {
  position: absolute;
  opacity: 0;
  left: 330px;
  bottom: 5px;
  font-size: 15pt;
}
.photo:hover #click {
  opacity: 1;
```

透明度が0に設定。
マウスを乗せると、透明度を1に変え、文字が現れるようにする。

メインページに掲示板を簡略に表示

3 きらめきエフェクト

```
.photo .glow-wrap {  
    overflow: hidden;  
    position: absolute;  
    width: 100%;  
    height: 100%;  
    top: 0;  
}  
  
.photo .glow {  
    display: block;  
    position: absolute;  
    width: 40%;  
    height: 200%;  
    background: rgba(255, 255, 255, .2);  
    top: 0;  
    filter: blur(5px); /*ガウシアン ブラ フィルタ*/  
    transform: rotate(45deg) translate(-450%, 0);  
    transition: all .5s cubic-bezier(0.645, 0.045, 0.355, 1);  
    /*3次 ベジエ 曲線(アニメーション 運動の調整)*/  
}
```

45度だけ回転し、左から右へ移動。

```
.photo:hover .glow {  
    transform: rotate(45deg) translate(450%, 0);  
    transition: all 1s cubic-bezier(0.645, 0.045, 0.355, 1);  
}
```

```
.photo:hover img, .photo:hover .glow-wrap {  
    margin-top: 0;  
}
```

글번호	카테고리	제목
16	자유게시판	アイサツノウタ
12	후기게시판	방편지
7	후기게시판	제목
8	자유게시판	제목
5	후기게시판	제목

各掲示板ごとに最新のボードを5つまで表示してくれるsql文を作成

1 全てのボード

```
String sql = "select B_NO, B_CATE, B_TITLE from (select * from BOARD_DB order by B_DATE desc)  
+ \"where rownum <= 5\";
```

2 フリー ボード

```
String sql = "select B_NO, B_CATE, B_TITLE from (select * from BOARD_DB where B_CATE like '자유게시판'  
+ \"order by B_DATE desc) where rownum <= 5";
```

3 レビュー ボード

```
String sql = "select B_NO, B_CATE, B_TITLE from (select * from BOARD_DB where B_CATE like '후기게시판'  
+ \"order by B_DATE desc) where rownum <= 5";
```

カテゴリー「ホーム」（担当：イム・ドンファン）

担当機能

- ホームエリアに反応型スライドとイメージマップ機能追加

- 1.自動的にスライドが変わる時、一緒にイメージマップのイメージも交換できるように具現
 - 2.イメージマップの特定区域をクリックすると該当するスライドに移動
 - 3.ウェブブラウザの幅が縮小されたらそれとともにスライドとイメージマップも一緒に減り、特定の幅になるとイメージマップを隠してボタンを表す（ボタンをクリックするとイメージマップを再び表示）
 - 4.イメージマップに「x」ボタンを追加し、ユーザーが自由にイメージマップを開閉できるように具現
 - 5.スライドを「aタグ」で包み込み、スライドをクリックするとそのスライド情報ページに移るように具現
- メンバーたちが作った掲示板、ウィジェットなどに反応型機能を適用し、ウェブブラウザが減る時一緒に減るように具現

使用機能・ツール

- Eclipse(html, JSP, JS, AJAX, CSS ...)

Bxslider jQueryを使ってスライド機能を実現 レイアウトを応用してデザインを変更

スライド機能

```
<div style="position: relative; height: 100%;" id="contain">
<!-- 슬라이더 영역 -->
<div class="bxslider" id="slider">
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=117"></a>
</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=8"></a>
</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=18"></a>
</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=2"></a>
</div>
<div>

</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=77"></a>
</div>
<div>

</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=72"></a>
</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=12"></a>
</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=73"></a>
</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=61"></a>
</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=22"></a>
</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=75"></a>
</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=74"></a>
</div>
<div>

</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=1"></a>
</div>
<div>

</div>
<div>

</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=120"></a>
</div>
<div>

</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=59"></a>
</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=4"></a>
</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=76"></a>
</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=11"></a>
</div>
<div>

</div>
<div>
<a href="http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=43"></a>
</div>
<div>
<!-- bxslider 닫음 -->
```

```
<!-- bxslider설정 -->
<script>
let slider = $('.bxslider').bxSlider({
    auto : true,
    autoControls : false,
    stopAutoOnClick : false,
    pager : false,
    controls : false,
    adaptiveHeight : true,
    touchEnabled :false
});
</script>
```

```
5 /* スライド テ두리 제거 */
6 .bx-wrapper {
7     border: 0;
8     background: #fbfbfb;
9     box-shadow: none;
10 }
11 }
```



スライド機能

```
<!-- 100 영역 -->


![자사구를 관리축제정보](dh_area_img/서울지도.png)

<area shape="poly"
coords="226,17,217,19,203,44,207,57,215,68,227,75,240,84,251,85,265,75,241,49,233,32"
href="javascript:custom_function(0); alt="강북구">
<area shape="poly"
coords="240,29,227,13,246,10,268,26,272,55,268,64,245,52,237,41"
href="javascript:custom_function(1); alt="은평구">
<area shape="poly"
coords="269,13,284,7,296,4,302,10,311,15,307,20,309,29,306,40,309,52,323,55,324,65,310,76,280,79,260,68,273,58,271,41"
href="javascript:custom_function(2); alt="노원구">
<area shape="poly"
coords="160,47,172,41,183,47,195,63,177,71,174,85,165,92,153,101,146,109,132,114,117,106,141,55"
href="javascript:custom_function(3); alt="은평구">
<area shape="poly"
coords="132,118,147,125,154,132,163,136,186,133,191,125,178,117,182,99,173,89"
href="javascript:custom_function(4); alt="시대문구">
<area shape="poly"
coords="178,73,195,67,202,75,206,83,206,98,201,97,242,122,212,123,195,124,181,120,185,107,179,87"
href="javascript:custom_function(5); alt="종로구">
<area shape="poly"
coords="205,59,239,87,270,83,286,88,265,96,255,105,242,111,228,106,204,99,213,93,198,66"
href="javascript:custom_function(6); alt="성북구">
<area shape="poly"
coords="244,115,248,122,261,120,287,131,293,121,286,105,286,89"
href="javascript:custom_function(7); alt="동대문구">
<area shape="poly"
coords="286,81,305,79,332,79,331,92,328,102,320,112,317,119,295,121,288,103"
href="javascript:custom_function(8); alt="용산구">
<area shape="poly"
coords="94,119,143,148,155,148,164,149,170,156,175,155,187,143,185,135,161,137,147,125,129,119,112,105"
href="javascript:custom_function(9); alt="마포구">
<area shape="poly"
coords="173,157,179,167,205,173,220,167,229,159,234,153,222,145,205,139,192,139"
href="javascript:custom_function(10); alt="용산구">
<area shape="poly"
coords="168,134,196,125,207,127,220,125,242,125,242,134,229,144"
href="javascript:custom_function(11); alt="종로구">
<area shape="poly"
coords="230,146,236,150,246,148,277,155,288,134,262,123,246,123,243,133"
href="javascript:custom_function(12); alt="성북구">
<area shape="poly"
coords="277,157,302,163,324,145,327,134,317,131,319,122,294,123"
href="javascript:custom_function(13); alt="은평구">
<area shape="poly"
coords="1,138,23,148,29,156,41,151,54,151,60,145,67,149,73,162,93,161,93,149,94,141,108,143,73,120,40,100,25,119"
href="javascript:custom_function(14); alt="강북구">
<area shape="poly"
coords="61,149,165,68,163,58,173,188,63,181,67,186,78,181,92,181,97,184,109,175,108,166,117,159,111,146,97,144,94,162,72,164"
href="javascript:custom_function(15); alt="양천구">
<area shape="poly"
coords="48,190,56,200,53,207,48,214,52,217,62,214,71,213,79,215,81,207,98,197,109,201,121,208,127,208,131,205,126,199,121,186,112,175,102,183,95,186,85,182,68,187,58,182"
href="javascript:custom_function(16); alt="관악구">
<area shape="poly"
coords="122,181,190,202,140,193,148,188,154,175,167,170,159,182,140,150,109,140,117,157,110,166,114,175"
href="javascript:custom_function(17); alt="영등포구">
<area shape="poly"
coords="101,202,113,218,115,230,122,234,127,242,127,249,139,248,147,245,155,236,150,229,141,230,141,221,136,216,140,211,134,207,124,210"
href="javascript:custom_function(18); alt="금천구">
<area shape="poly"
coords="135,203,153,197,156,193,161,197,181,195,186,194,188,201,194,209,203,209,203,199,205,189,198,184,173,174,158,178,150,188"
href="javascript:custom_function(19); alt="서초구">
<area shape="poly"
coords="156,205,142,211,139,216,143,225,150,227,156,234,158,240,165,248,175,246,188,245,191,239,199,239,203,229,209,229,209,220,201,212,188,206,184,195,162,197,156,195"
href="javascript:custom_function(20); alt="관악구">
<area shape="poly"
coords="207,190,205,213,210,218,211,227,228,224,217,235,231,250,224,253,219,255,227,257,244,267,249,291,248,299,242,309,228,300,213,271,220,261,207,252,202,236,169"
href="javascript:custom_function(21); alt="서초구">
<area shape="poly"
coords="236,166,254,200,261,202,271,219,288,214,301,210,311,224,323,224,331,227,339,222,320,198,291,189,283,172,272,164,249,158"
href="javascript:custom_function(22); alt="강남구">
<area shape="poly"
coords="286,175,291,188,314,194,321,198,339,220,344,214,350,214,362,204,372,198,372,198,366,185,368,189,353,183,352,170,333,165,335,156,330,152,318,168,305,173"
href="javascript:custom_function(23); alt="송파구">
<area shape="poly"
coords="331,143,337,153,335,163,168,357,172,373,148,386,147,393,146,391,136,385,114,361,126,344,129"
href="javascript:custom_function(24); alt="강동구">


```

```
<map name="autonomyMap">
<area shape="poly"
coords="226,17,217,19,203,44,207,57,215,68,227,75,240,84,251,85,265,75,241,49,233,32"
href="javascript:custom_function(0); alt="강북구">
```



区別に座標を設定して区域を指定し、
該当区域をクリックすると、
Javascript:custom_function(n);関数実行

スライド機能

```

<script>
function custom_function(index) {
    if (typeof slider.goToSlide === 'function') {
        slider.goToSlide(index); // 해당 슬라이더로 이동
    } else { // 무엇인가 잘못되었으면
        alert('에러!');
    }

    if (index == 0) {
        document.getElementById("imgId1").src = "dh_area_img/강북구.png";
    }
    if (index == 1) {
        document.getElementById("imgId1").src = "dh_area_img/도봉구.png";
    }
    if (index == 2) {
        document.getElementById("imgId1").src = "dh_area_img/노원구.png";
    }
    if (index == 3) {
        document.getElementById("imgId1").src = "dh_area_img/은평구.png";
    }
    if (index == 4) {
        document.getElementById("imgId1").src = "dh_area_img/서대문구.png";
    }
    if (index == 5) {
        document.getElementById("imgId1").src = "dh_area_img/종로구.png";
    }
    if (index == 6) {
        document.getElementById("imgId1").src = "dh_area_img/성북구.png";
    }
    if (index == 7) {
        document.getElementById("imgId1").src = "dh_area_img/동대문구.png";
    }
    if (index == 8) {
        document.getElementById("imgId1").src = "dh_area_img/중랑구.png";
    }
    if (index == 9) {
        document.getElementById("imgId1").src = "dh_area_img/마포구.png";
    }
    if (index == 10) {
        document.getElementById("imgId1").src = "dh_area_img/용산구.png";
    }
    if (index == 11) {
        document.getElementById("imgId1").src = "dh_area_img/중구.png";
    }
    if (index == 12) {
        document.getElementById("imgId1").src = "dh_area_img/성동구.png";
    }
    if (index == 13) {
        document.getElementById("imgId1").src = "dh_area_img/광진구.png";
    }
    if (index == 14) {
        document.getElementById("imgId1").src = "dh_area_img/강서구.png";
    }
    if (index == 15) {
        document.getElementById("imgId1").src = "dh_area_img/양천구.png";
    }
    if (index == 16) {
        document.getElementById("imgId1").src = "dh_area_img/구로구.png";
    }
    if (index == 17) {
        document.getElementById("imgId1").src = "dh_area_img/영등포구.png";
    }
    if (index == 18) {
        document.getElementById("imgId1").src = "dh_area_img/금천구.png";
    }
    if (index == 19) {
        document.getElementById("imgId1").src = "dh_area_img/동작구.png";
    }
    if (index == 20) {
        document.getElementById("imgId1").src = "dh_area_img/관악구.png";
    }
    if (index == 21) {
        document.getElementById("imgId1").src = "dh_area_img/서초구.png";
    }
    if (index == 22) {
        document.getElementById("imgId1").src = "dh_area_img/강남구.png";
    }
    if (index == 23) {
        document.getElementById("imgId1").src = "dh_area_img/송파구.png";
    }
    if (index == 24) {
        document.getElementById("imgId1").src = "dh_area_img/강동구.png";
    }
}
</script>

```

イメージマップ生成時、区域別にindexを使って割り当てる。そして各イメージマップをクリックすると、該当するスライダーに移動するように具現



ドボン区スライドに移動



```

if (index == 0) {
    document.getElementById("imgId1").src = "dh_area_img/강북구.png";
}
if (index == 1) {
    document.getElementById("imgId1").src = "dh_area_img/도봉구.png";
}
if (index == 2) {
    document.getElementById("imgId1").src = "dh_area_img/노원구.png";
}
if (index == 3) {
    document.getElementById("imgId1").src = "dh_area_img/은평구.png";
}
if (index == 4) {
    document.getElementById("imgId1").src = "dh_area_img/서대문구.png";
}
if (index == 5) {
    document.getElementById("imgId1").src = "dh_area_img/종로구.png";
}

```

ドボン区クリック時



スライド機能

JavaScriptに「for文」を使用してスライドが自動的に変わることなく、たびにi値を付与し、クリックしなくても該当するイメージマップに交替できるよう具現

```
<script type="text/javascript">
$(function() {
    var observer = new MutationObserver(
        function(mutations) {
            target = document.getElementById('slider');
            let child = $(target).find("div");
            for (var i = 0; i < 26; i++) {
                let includeFalse = $(child[i]).attr("aria-hidden");
                if (includeFalse == 'false') {
                    /* alert(i); */

                    if (i == 1) {
                        document.getElementById("imgId1").src = "dh_area_img/강북구.png";
                    }
                    if (i == 2) {
                        document.getElementById("imgId1").src = "dh_area_img/도봉구.png";
                    }
                    if (i == 3) {
                        document.getElementById("imgId1").src = "dh_area_img/노원구.png";
                    }
                    if (i == 4) {
                        document.getElementById("imgId1").src = "dh_area_img/은평구.png";
                    }
                    if (i == 5) {
                        document.getElementById("imgId1").src = "dh_area_img/서대문구.png";
                    }
                }
            }
        }
    );
});
```

Ex) i==2の時



Ex) $i==3$ の時



反応型ウェブブラウザ

```
19  /* 반응형 코드 시작 */
20
21 @media ( min-width : 1024px ) {
22@  #MainContainer{
23    display: flex;
24    position: relative;
25    width: 100%;
26    height: 600px;
27    /* border: 5px solid blue; */
28 }
29
30@ #leftInformation {
31  /* border : 5px solid red; */
32  width: 40%;
33  height: 100%;
34  display: flex;
35  flex-direction: column;
36  position: absolute;
37 }
38@ #calendar {
39  font-size: 13pt;
40  position : absolute;
41  width: 50%;
42  height:102%;
43  right: 0px;
44 }
```

```
120 @media(max-width : 1023px){
121
122@ #MainContainer {
123  display: flex;
124  position: relative;
125  width: 100%;
126  height: 460px;
127  /* border: 5px solid blue; */
128 }
129
130
131@ #leftInformation {
132  width: 40%;
133  height: 100%;
134  font-size: 80%;
135  position: absolute;
136 }
137
138@ #calendar {
139  font-size: 11pt;
140  position: absolute;
141  width: 50%;
142  height: 100%;
143  right: 0px;
144 }
145
```

```
267 @media ( max-width : 900px ) {
268
269@ #MainContainer {
270  display: flex;
271  position: relative;
272  width: 100%;
273  height: 422px;
274  /* border: 5px solid blue; */
275 }
276@ #leftInformation {
277  width: 40%;
278  font-size: 60%;
279  position: absolute;
280 }
281@ #calendar {
282  font-size: 9pt;
283  position: absolute;
284  width: 50%;
285  height: 100%;
286  right: 0px;
287 }
288
289@ #img_content {
```

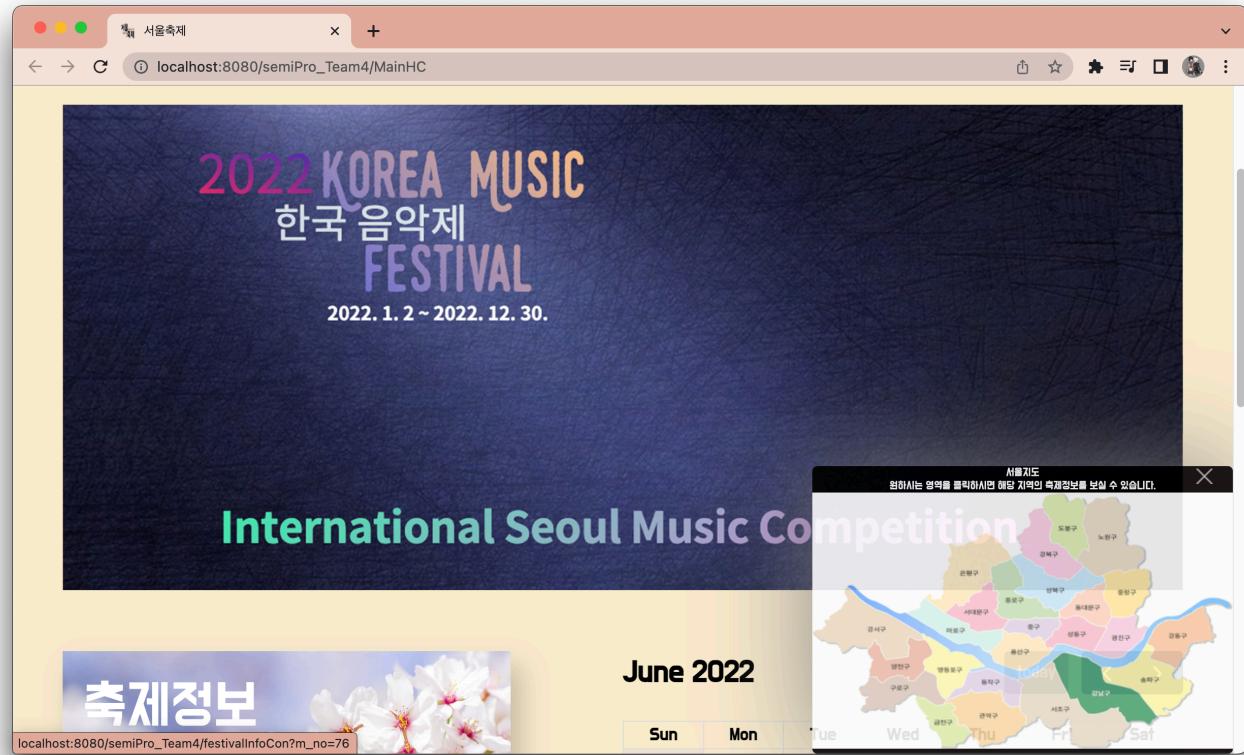
```
525 @media ( max-width : 700px ) {
526
527@ #MainContainer {
528  position: relative;
529  min-width: 300px;
530  height: 1000px;
531  margin-left: auto;
532  margin-right: auto;
533 }
534@ #leftInformation {
535  height: 60%;
536  width: 100%;
537  font-size: 80%;
538  position: absolute;
539 }
540@ #calendar {
541  top: 65%;
542  height: 35%;
543  width: 100%;
544  position: absolute;
545 }
```

```
395 @media ( max-width : 800px ) {
396
397@ #MainContainer {
398  display: flex;
399  position: relative;
400  width: 100%;
401  height: 389px;
402  /* border: 5px solid blue; */
403 }
404@ #leftInformation {
405  width: 40%;
406  font-size: 50%;
407  position: absolute;
408 }
409@ #calendar {
410  font-size: 7pt;
411  position: absolute;
412  width: 50%;
413  height: 100%;
414  right: 0px;
415 }
416
```

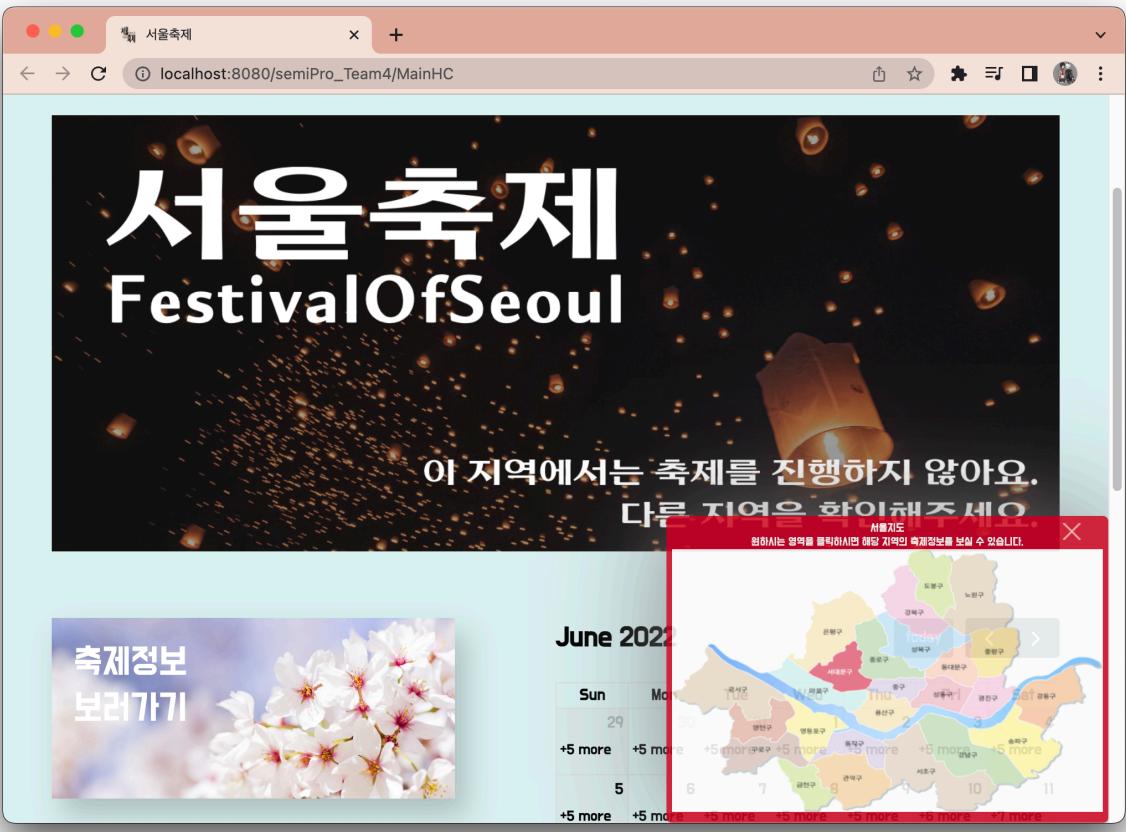
MediaQueryを使用して
ウェブブラウザのwidthによっ
て、
それぞれの値を設定して反応型
ウェブを製作

反応型ウェブブラウザ

ウェブブラウザmin-width:1024px(default)



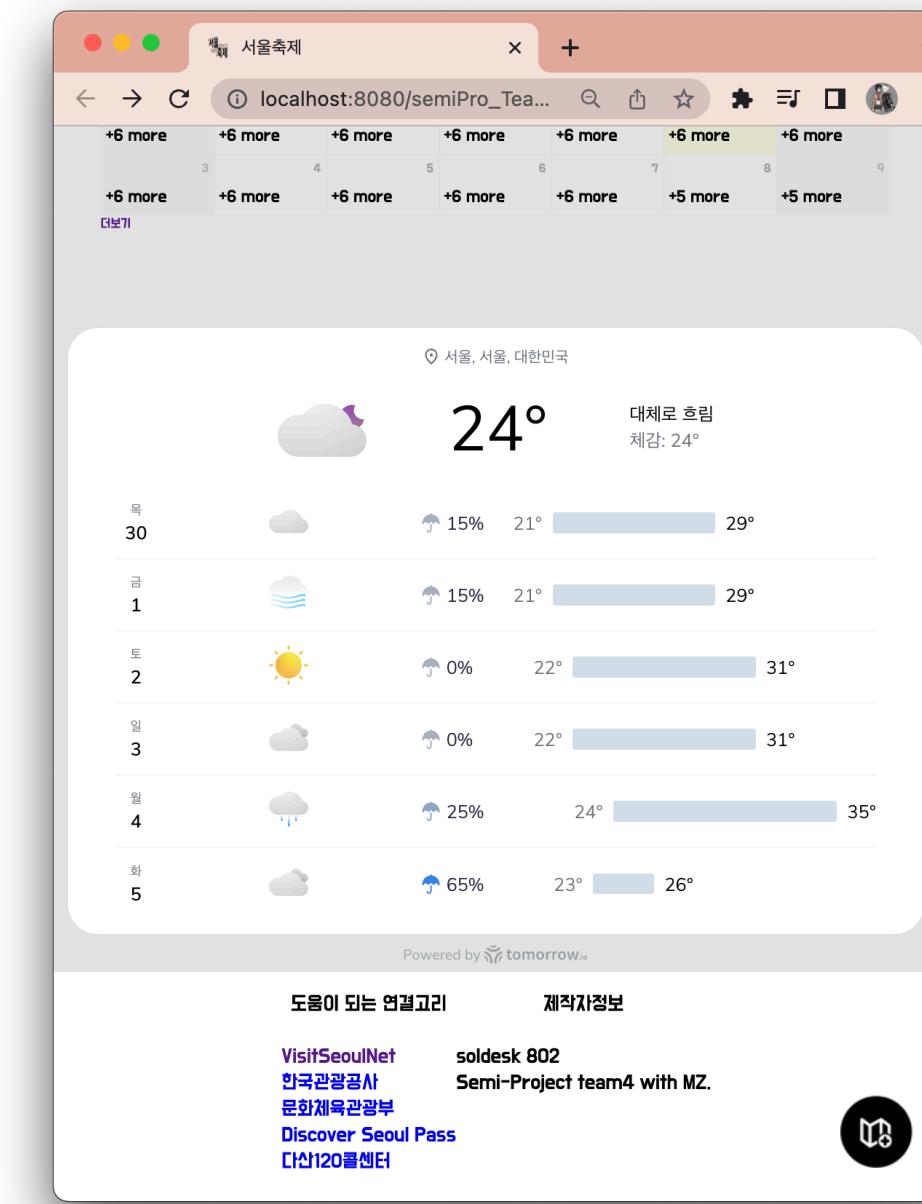
ウェブブラウザmax-width:1023px



ウェブブラウザmax-width:900px



ウェブブラウザmax-width:700px(モバイルver.)



ウェブブラウザ
max-width:800px



カテゴリ 「マップ」(担当: パク・シウォン)

担当機能

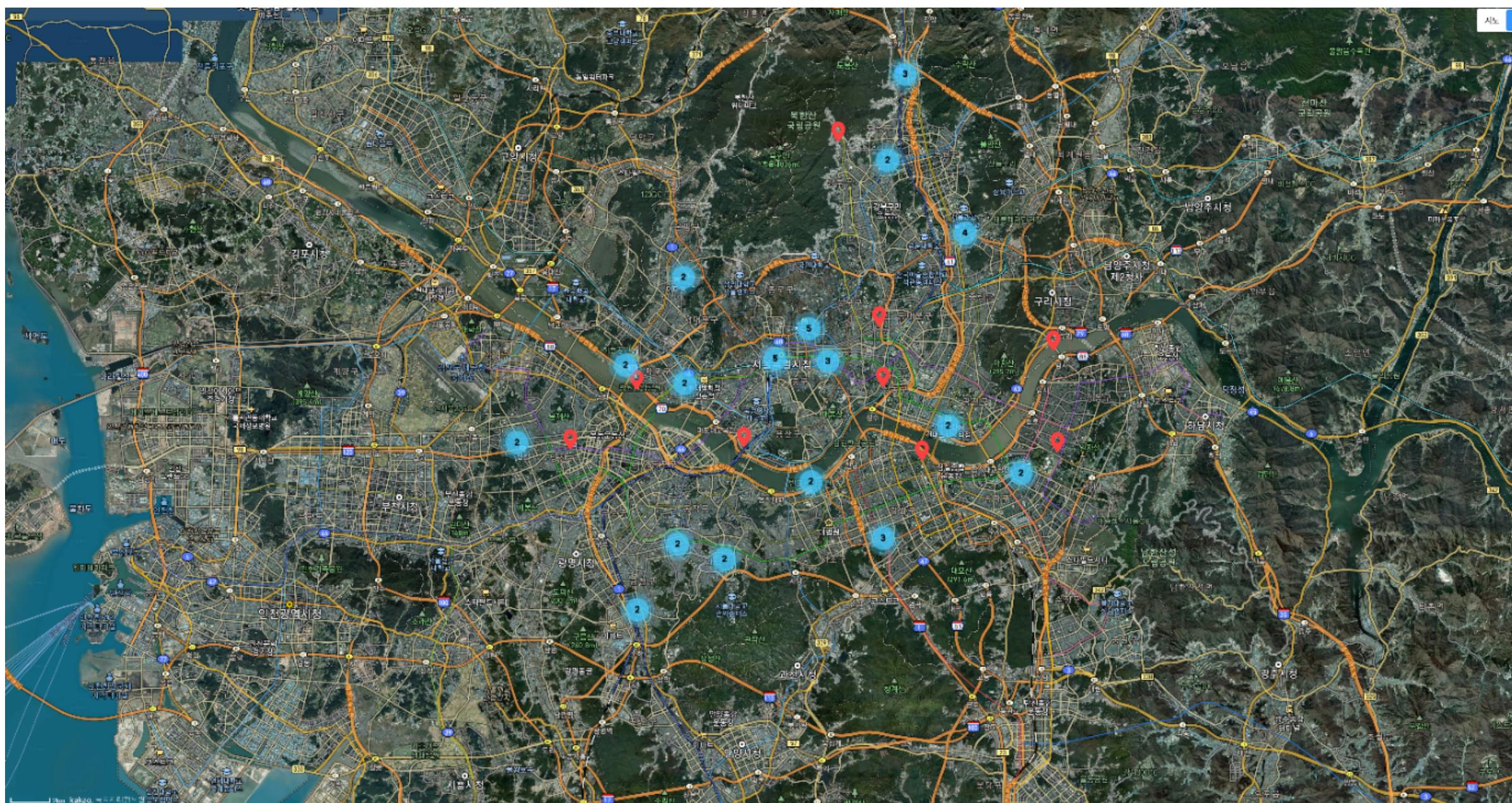
カテゴリー「マップ」

1. カカオマップAPI連動及び地図作成
2. クリックで作動する関数が重複しないように条件文を通じて制御
3. クラスタリング機能(zoom levelによって近接距離のマーカーを一つのグループとして扱う)
4. マーカー生成(jsonデータの情報を地図上に表示)
5. マーカーにマウスを乗せる時だけインプットウィンドウが見えるように設定
6. googlemapカスタムオーバーレイを使用して、マーカーをクリックする時にお祭り情報のオーバーレイ表示
7. ユーザーの現在位置情報を追跡(Geocoder、オブジェクトを使用)して地図の機能を具現・提供
8. カテゴリ「マップ」ページCSS

使用機能/ツール

- eclipse(html, jsp, js, jquery)
- Oracle SQL Developer

地図生成及び座標データ地図上に表示



```
<div id="map" style="width:100%;height:95vh;"></div>
<script type="text/javascript" src="jQuery.js"></script>
<script type="text/javascript" src="//dapi.kakao.com/v2/maps/sdk.js?appkey=e8937aa7ae89b7653202e90fcc96c05&libraries=service,clusterer"></script>
<script src="kakaoMapsJavaScriptAPIwrapper.js"></script>
<script src="http://code.jquery.com/jquery-1.9.1.min.js"></script>
<script>

var map = new kakao.maps.Map(document.getElementById('map'), { // 지도를 표시할 div
    center : new kakao.maps.LatLng(37.542223,126.9803401), // 지도의 중심좌표
    level : 9 // 지도의 확대 레벨
});

// 마커 클러스터러를 생성
// 마커 클러스터러를 생성할 때 disableClickZoom 값을 true로 지정하지 않은 경우
// 클러스터 마커를 클릭했을 때 클러스터 객체가 포함하는 마커들이 모두 잘 보이도록 지도의 레벨과 영역을 변경
// 이 예제에서는 disableClickZoom 값을 true로 설정하여 기본 클릭 동작을 막고
// 클러스터 마커를 클릭했을 때 클릭된 클러스터 마커의 위치를 기준으로 지도를 1레벨씩 확대
var clusterer = new kakao.maps.MarkerClusterer({
    map: map, // 마커들을 클러스터로 관리하고 표시할 지도 객체
    averageCenter: true, // 클러스터에 포함된 마커들의 평균 위치를 클러스터 마커 위치로 설정
    minLevel: 5, // 클러스터 할 최소 지도 레벨
    disableClickZoom: true // 클러스터 마커를 클릭했을 때 지도가 확대되지 않도록 설정
});

// 데이터를 가져오기 위해 jQuery를 사용
// 데이터를 가져와 마커를 생성하고 클러스터러 객체에 넘겨줌

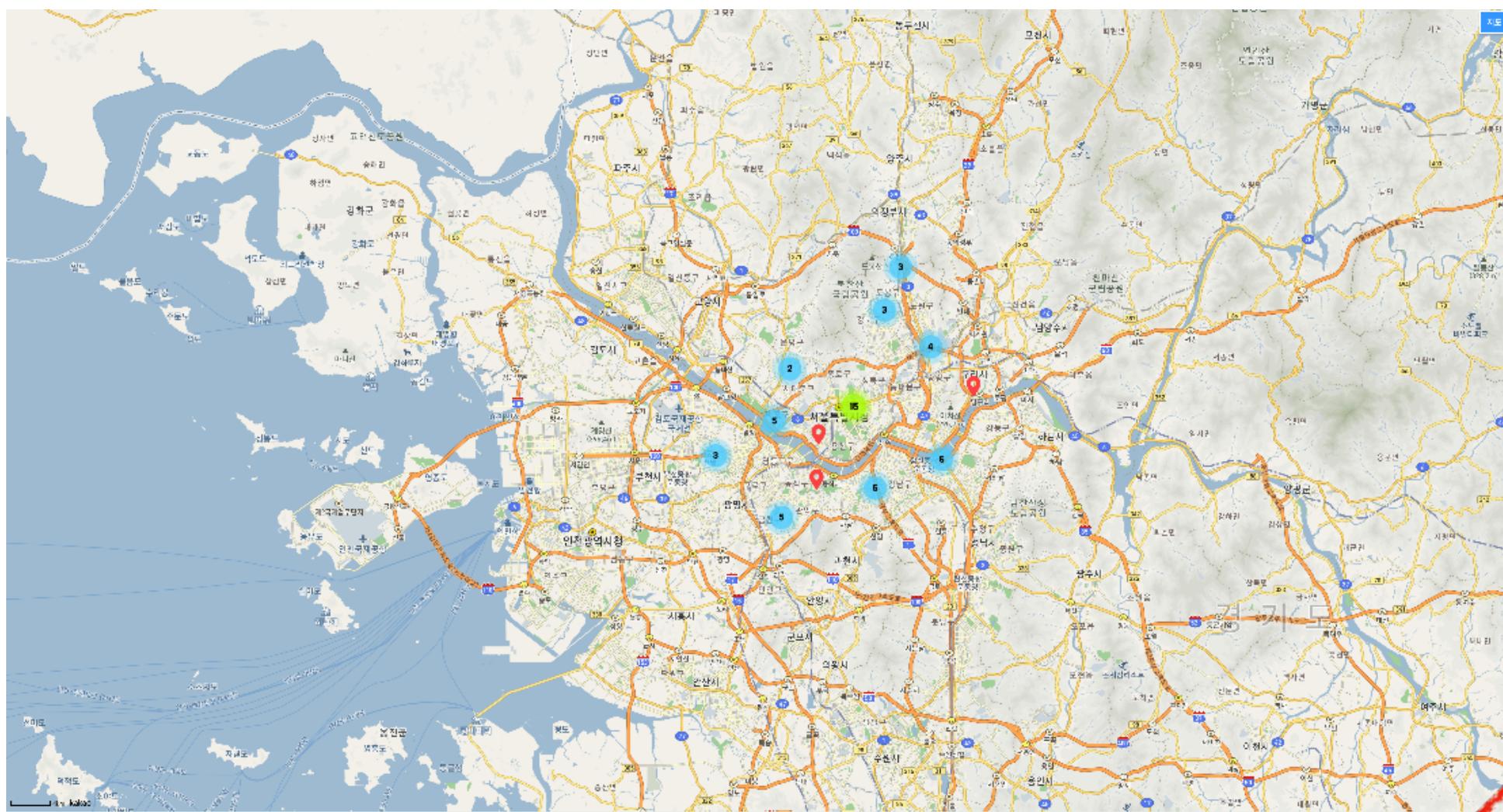
var imageSrc = 'SW_image/address-location-map-marker-red-icon-26350.png', // 마커이미지의 주소
imageSize = new kakao.maps.Size(44, 49), // 마커이미지의 크기입니다
imageOption = {offset: new kakao.maps.Point(27, 69)}; // 마커이미지의 옵션. 마커의 좌표와 일치시킬 이미지 안에서의 좌표를 설정

// 마커의 이미지정보를 가지고 있는 마커이미지를 생성합니다
var markerImage = new kakao.maps.MarkerImage(imageSrc, imageSize, imageOption),
markerPosition = new kakao.maps.LatLng(37.54699, 127.09598); // 마커가 표시될 위치

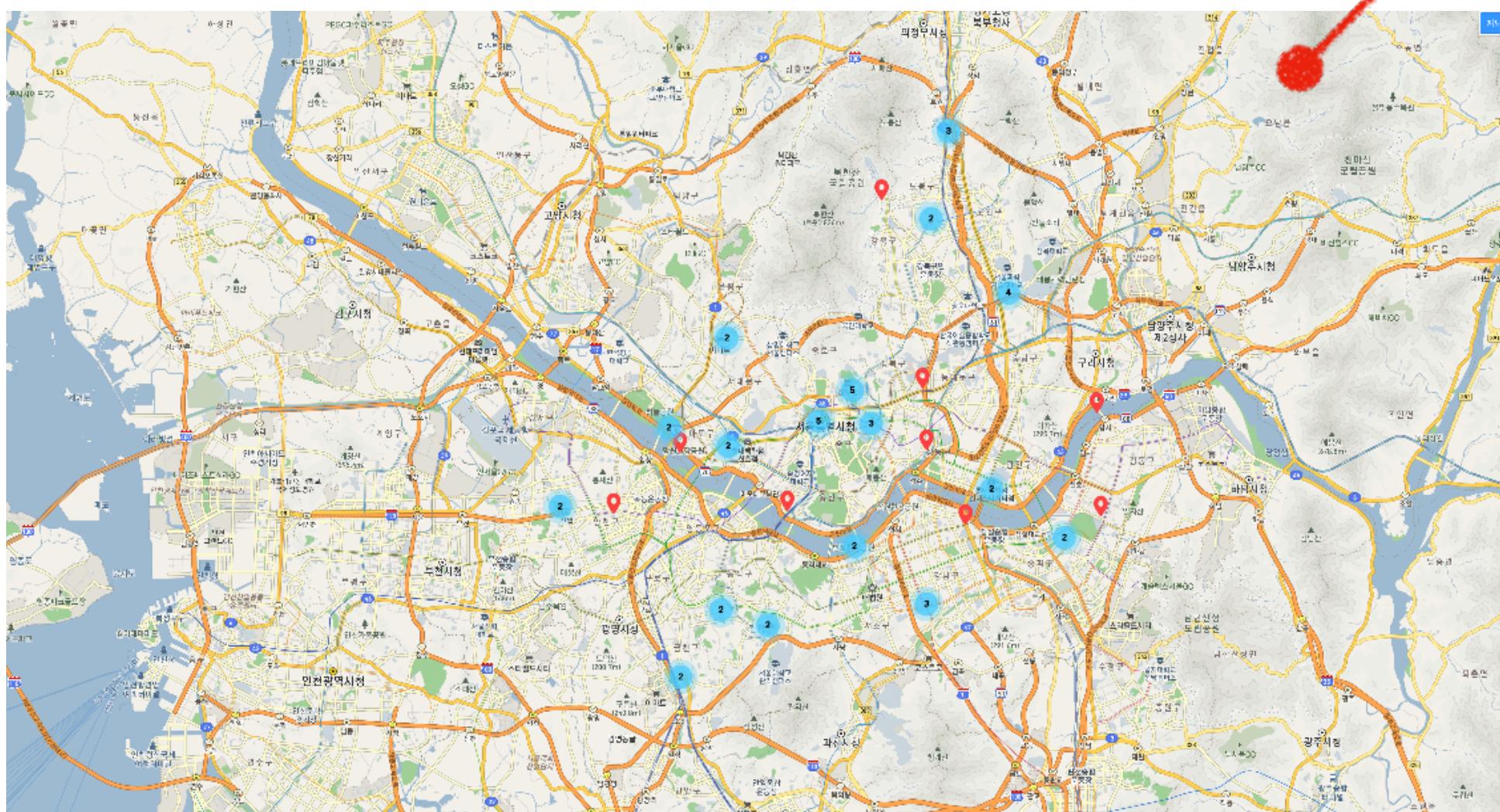
$.get("SW_mapFolder/mapdata.json", function(data) {
    // 데이터에서 좌표 값을 가지고 마커를 표시
    // 마커 클러스터러로 관리할 마커 객체는 생성할 때 지도 객체를 설정하지 않음

    var markers = $(data.positions).map(function(i, position) {
        var marker = new kakao.maps.Marker({
            position : new kakao.maps.LatLng(position.lat, position.lng),
            image: markerImage
        });
    });
});
```

クラスタリング (click)



クリックすると1レベルずつズームイン



```
// 마커 클러스터에 클릭이벤트를 등록  
// 마커 클러스터를 생성할 때 disableClickZoom을 true로 설정하지 않은 경우  
// 이벤트 핸들러로 cluster 객체가 넘어오지 않을 수도 있음
```

```
kakao.maps.event.addListener(clusterer, 'clusterclick', function(cluster) {  
  
    // 현재 지도 레벨에서 1레벨 확대한 레벨  
    var level = map.getLevel()-1;  
  
    // 지도를 클릭된 클러스터의 마커의 위치를 기준으로 확대  
    map.setLevel(level, {anchor: cluster.getCenter()});  
});
```

```
// 위치정보 파악 가능  
var geocoder = new kakao.maps.services.Geocoder();  
function searchDetailAddrFromCoords(coords, callback) {  
    // 좌표로 법정동 상세 주소 정보를 요청  
    geocoder.coord2Address(coords.getLng(), coords.getLat(), callback);  
}
```

```
// 지도 타입 변경 컨트롤을 생성  
var mapTypeControl = new kakao.maps.MapTypeControl();
```

```
// 지도의 상단 우측에 지도 타입 변경 컨트롤을 추가  
map.addControl(mapTypeControl, kakao.maps.ControlPosition.TOPRIGHT);
```

```
// 지도에 확대 축소 컨트롤을 생성  
var zoomControl = new kakao.maps.ZoomControl();
```

```
// 지도의 우측에 확대 축소 컨트롤을 추가  
map.addControl(zoomControl, kakao.maps.ControlPosition.RIGHT);
```

インフォウィンドウ (mouseover)



```
var infowindow = new kakao.maps.InfoWindow({
    content: "<div class='infocontent'>" + position.title + "</div>" // 인포윈도우에 표시할 내용
});
kakao.maps.event.addListener(marker, 'mouseover', makeOverListener(map, marker, infowindow));
kakao.maps.event.addListener(marker, 'mouseout', makeOutListener(infowindow));
```

```
function makeOverListener(map, marker, infowindow) {
    return function() {
        infowindow.open(map, marker);
    };
}
var add = position.place,
    // 인포윈도우를 닫는 클로저를 만드는 함수
function makeOutListener(infowindow) {
    return function() {
        infowindow.close();
    };
}
function closeOverlay() {
    overlay.close(null);
}
```

// 마커 위에 커스텀오버레이를 표시
// 다른 오버레이창 클릭시 기존에 열려있는 오버레이 닫아주는 기능구현

```
var overlay = new kakao.maps.CustomOverlay({
    content: content,
    map: map,
    title: position.place,
    position: marker.getPosition(),
    disableClickZoom: true,
});

if (clickedOverlay) {
    clickedOverlay.setMap(null);
}

overlay.setMap(map);
clickedOverlay = overlay;
// 전역 변수를 하나 두고 클릭할 때마다 이 변수에 할당되어 있는 CustomOverlay를 setMap(null) 하고
// 현재 클릭된 CustomOverlay를 해당 변수에 재할당하여 계속하여 이 전 CustomOverlay를 지우는 방식

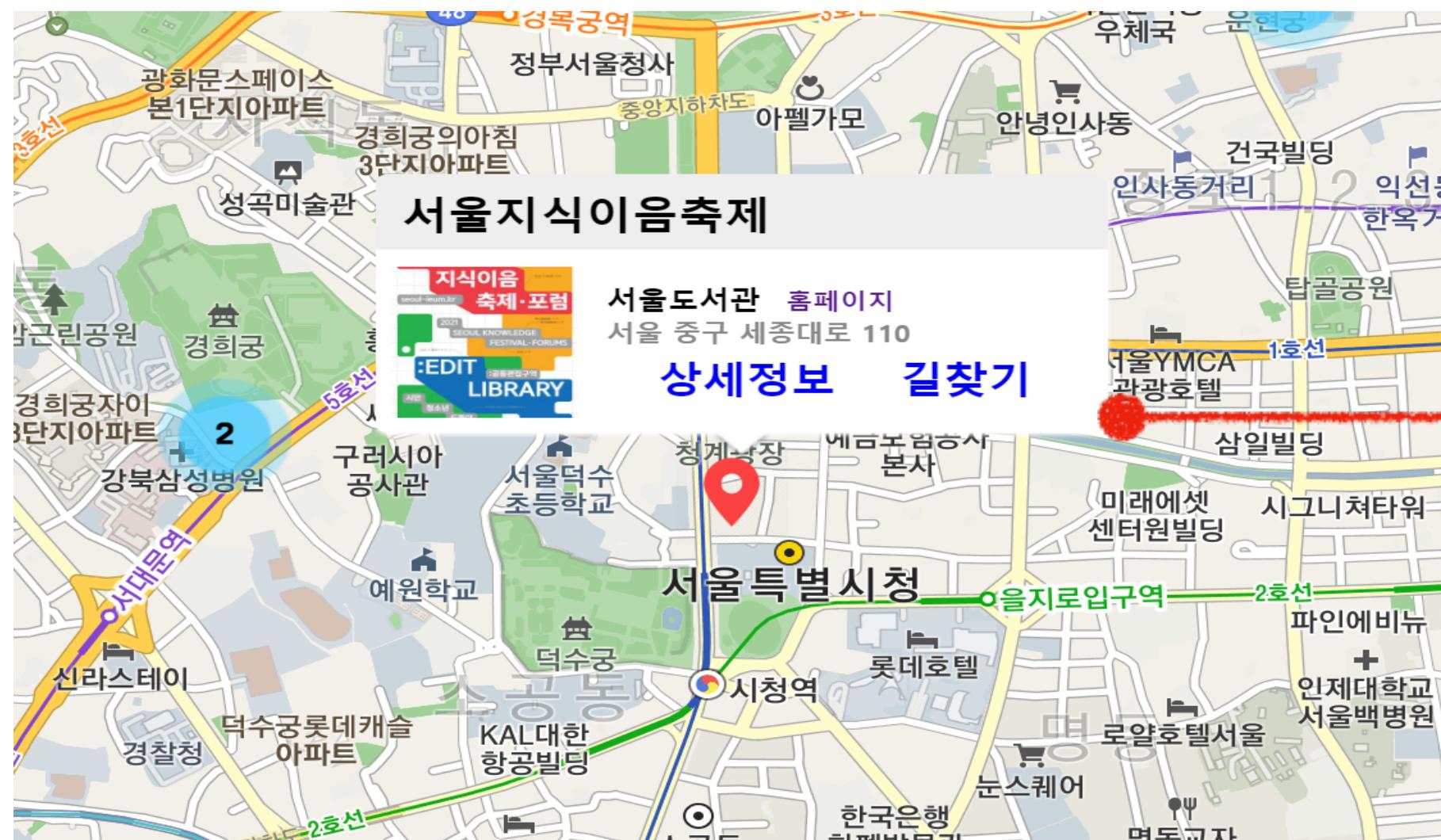
//location.href = "https://map.kakao.com/?sName="+detailAddr+"&eName="+position.place
};

return marker;
});

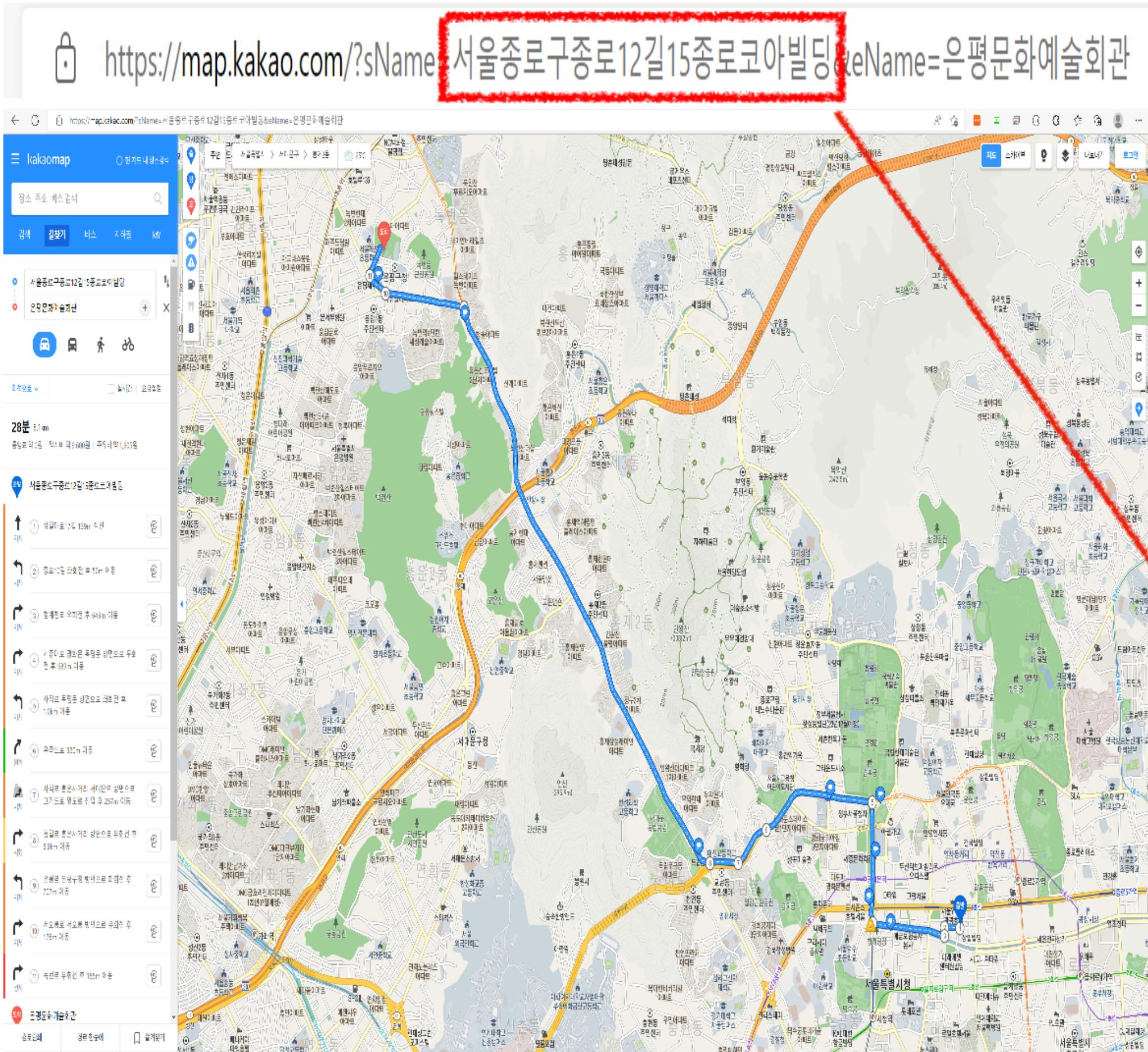
// 클러스터리에 마커들을 추가
clusterer.addMarkers(markers);
marker.setMap(map);

});
```

カスタムオーバレイ (click)



道を検索



```

var detailAddr; // 위치 검색 후 받을 변수 설정
// HTML5의 geolocation으로 사용할 수 있는지 확인
if (navigator.geolocation) {
    // GeoLocation을 이용해서 접속 위치를 얻어온다.
    navigator.geolocation.getCurrentPosition(function(position){
        var lat = position.coords.latitude, // 위도
            lon = position.coords.longitude; // 경도

        var locPostion = new kakao.maps.LatLng(lat, lon), //마커가 표시될 위치를 geolocation 좌표로 생성
            message = '<div style="padding:5px;">여기에 계신가요?!</div>'; // 인포윈도우에 표시될 내용

        // 주소-좌표 변환 객체를 생성합니다
        searchDetailAddrFromCoords(locPostion, function(result, status) {
            if (status === kakao.maps.services.Status.OK) {
                detailAddr = !!result[0].road_address ? result[0].road_address.address_name : result[0].address.address_name;
                //location.href = "https://map.kakao.com/?sName="+detailAddr+"&eName="+"??" //???값은 position.place */
            }
        });
    });
}

erlay=null;
ickedOverlay=null;

마커에 클릭이벤트를 등록
kao.maps.event.addListener(marker, 'click', function () {
    detailAddr = detailAddr.replace(/ /g,""); //replace로 공백 제거

    var content = "<div class='wrap'>"+ "<div class='info'>"+ "<div class='title'>" + position.title + "<div onclick='closeOverlay()' title='닫기'>" + "</div>
<div class='body'>"+ "<div class='img'>" + "<img src='"+position.img+" width='73' height='70'" + "</div>
<div class='desc'>"+ "<div class='ellipsis'>" + position.place + "<span class='span3'>" + "<a href='"+position.hp + " target ='_blank'>" + "홈페이지" + "</a>" +
<div class='jibun ellipsis'>" + position.addr + "</div>
<div>"+ "<span class='span1'>" + "<a href='http://localhost:8080/semiPro_Team4/festivalInfoCon?m_no=' + position.m_no + ">" +
"상세정보" + "</a>" + "</span>" +
<span class='span2'>" + "<a href=https://map.kakao.com/?sName=" + detailAddr + "&eName=" + position.place + ">" + "길찾기" + "</a>" + "</span>" +
</div>" + "</div>" + "</div>" + "</div>";

alert(detailAddr);           //공백 포함 해서 나오므로 검색시 오류 (ex 부산 남구 80-1 oo아파트) //공백제거 해야함
alert(position.place);

```

10. 感想

パク・シウォン

今回のプロジェクトで私が担当したのは、マップカテゴリーです。お祭りが行われる場所が分かるマップを作り、ユーザーにより見やすく提供するのが目標でした。マップAPIの例文を参考にコードを作成し、具現したい機能を組み合わせました。コーディングすることにあたりエラーもいっぱい起こしましたが、チームの方々と先生のおかげでうまく解決することができました。具現したいことを具現することができて、やりがいを感じました。



イム・ドンフン

今回のセミプロジェクトでチームメンバーと協業してお互いにコミュニケーションをしながら意見を交わし、Githubの使い方を学ぶことができてとても良かったです。そして反応型スライドとイメージマップ、CSS作業をしながら新しい機能を学ぶことができて意味深い時間でした!これまで知っていた機能の応用法を身につき、問題が発生した時に開発者道具を通じて問題を解決しながらコード分析能力も成長したようでやりがいを感じました!



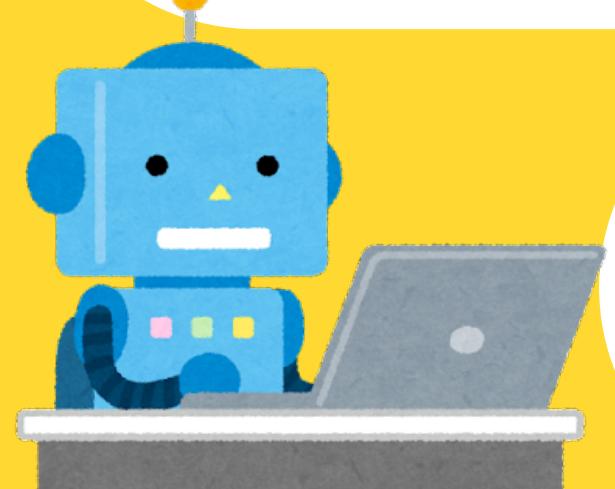
パク・ミンジ

私は祭りのAPIをベースして「祭り情報一覧」カテゴリーを作ることを担当しました。プロジェクトを進めながら知らない部分は検索を通じて一つずつ成功することに達成感を感じ、問題解決能力も大幅に上昇したようです。何よりMVCとSQL文、CSSなどを試みながら授業時間に学びましたが理解できなかった部分をこの機会に復習することができてよかったです。チームプロジェクトをしながら助言をくださった先生と辛い時、お互いに励まして導いてくれたメンバーたちにも感謝を申し上げます。



ジョン・ユジン

ホームページの全体CSSと掲示板機能を担当しました。今回のプロジェクトで想像した機能を実現させる能力の感覚を掴んだようで、とても嬉しいです。機能を実現しながら、プロジェクト後半になるほどプログラミングに自信がつきました。また、チームリーダーとして活動し、チームメンバーの力量とスケジュールを管理することの難しさと重要性についても知ることができました。間違いも多く、問題も多くの過程でしたが、最後までプロジェクトを完成することができてやりがいを感じました。



Q&A

氏名	田侑眞 (ジョン・ユジン)	朴旼智 (パク・ミンジ)	林東勲 (イム・ドンファン)	朴視願 (パク・シウォン)
e-mail	lita.y.jeon@gmail.com	busy8863@naver.com	dla_ehdgns@naver.com	soobin414@gmail.com
担当	チームリーダー デザイン/ CSS / 掲示板 / DB管理 / ウィジェットなど	CSS / 祭り情報パース/ (祭り情報一覧)/ DB管理など	CSS / ホームのスライダーなど	地図API適用 など



github

<https://github.com/LitaHart/SemiProject4Fes.git>

■
■