

The background of the slide features a complex network diagram. It consists of numerous nodes of varying sizes, some solid black, some solid blue, and some white with black outlines. These nodes are interconnected by a web of thin, light gray lines. The overall aesthetic is modern and technological, suggesting themes of data, connectivity, and networks.

공공데이터 활용 사례 분석 및 융합서비스 기획

공공데이터 활용 사례 분석 및 융합서비스 기획

심규영

목차

- 1) 공공 데이터 API 신청 및 확인
- 2) Kakao Maps API 신청 및 확인
- 3) 공공데이터를 활용한 서비스 개발
- 4) CODE

1) 공공 데이터 API 신청 및 확인

보건의료

자치행정기관

XML

JSON

부산광역시_동물병원 현황

부산광역시 관내 동물병원(동물병원명, 소재지, 전화번호, 인허가일자 등)에 대한 안내 및 정보를 제공하는 API 서비스

제공기관 부산광역시 수정일 2022-08-19 조회수 1032 활용신청 36 키워드 동물병원,반려동물,동물치료

데이터명	부산광역시_동물병원 현황 상세설명		
서비스유형	REST	심의여부	자동승인
신청유형	개발계정 활용신청	처리상태	승인
활용기간	2023-02-06 ~ 2025-02-06		

1) 공공 데이터 API 신청 및 확인

apis.data.go.kr/6260000/BusanAnimalHospService/getTblAnimalHospital?serviceKey=A3N1yO94YtYt

GET

URL params

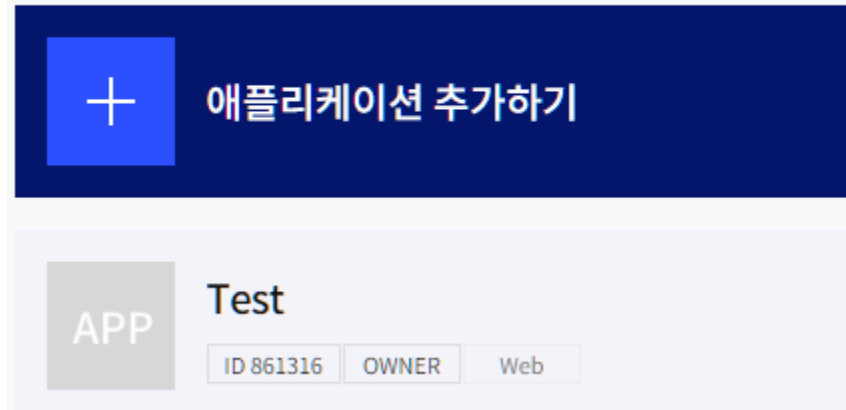
Headers (0)

serviceKey	A3N1yO94YtYVaTZIGtqnDtXz6H6jCzy	✕
pageNo	1	✕
numOfRows	10	✕
resultType	json	✕

```
{
  "getTblAnimalHospital": {
    "header": {
      "resultCode": "00",
      "resultMsg": "NORMAL_CODE"
    },
    "body": {
      "items": {
        "item": [
          {
            "gugun": "해운대구",
            "animal_hospital": "BS조은동물병원",
            "approval": "2011-02-10",
            "road_address": "부산광역시 해운대구 윗반송로 73 (반송동)",
            "tel": "051-544-7588",
            "lat": "35.230555",
            "lon": "129.156999",
            "basic_date": "2023-01-17"
          }
        ]
      }
    }
  }
}
```

2) KAKAO MAPS API 신청 및 확인

전체 애플리케이션 (1)



개발자 등록 및 앱 생성

Web

사이트 도메인

http://localhost:8080

• 카카오 로그인 사용 시 Redirect URI를 등록해야 합니다. [등록하러 가기](#)

웹 플랫폼 추가
사이트 도메인 등록

3) 공공데이터를 활용한 서비스 개발

프로젝트 개요	부산광역시 동물병원 API를 활용한 동물병원 Map 개발
프로젝트 이름	MapBusanAnimalHospital
Framework	Spring Boot 2.7.8
Build	Gradle – Groovy
Group	Map.busan.animal.hospital
Artifact	MapBusanAnimalHospital
Package	Map.busan.animal.hospital
Packaging	Jar
Language	Java 11
Dependencies	Spring Boot DevTools, Lombok, Spring Web, Thymeleaf

3) 공공데이터를 활용한 서비스 개발

Project

☒ Gradle - Groovy

☐ Gradle - Kotlin ☐ Maven

Spring Boot

☐ 3.0.3 (SNAPSHOT) ☐ 3.0.2 ☐ 2.7.9 (SNAPSHOT)

☒ 2.7.8

Project Metadata

Group map.busan.animal.hospital

Artifact MapBusanAnimalHospital

Name MapBusanAnimalHospital

Description Demo project for Spring Boot

Package name map.busan.animal.hospital

Packaging ☒ Jar ☐ War

Java ☐ 19 ☐ 17 ☒ 11 ☐ 8

Language

☒ Java ☐ Kotlin

☐ Groovy

Dependencies

ADD DEPENDENCIES... CTRL + B

Spring Boot DevTools DEVELOPER TOOLS

Provides fast application restarts, LiveReload, and configurations for enhanced development experience.

Lombok DEVELOPER TOOLS

Java annotation library which helps to reduce boilerplate code.

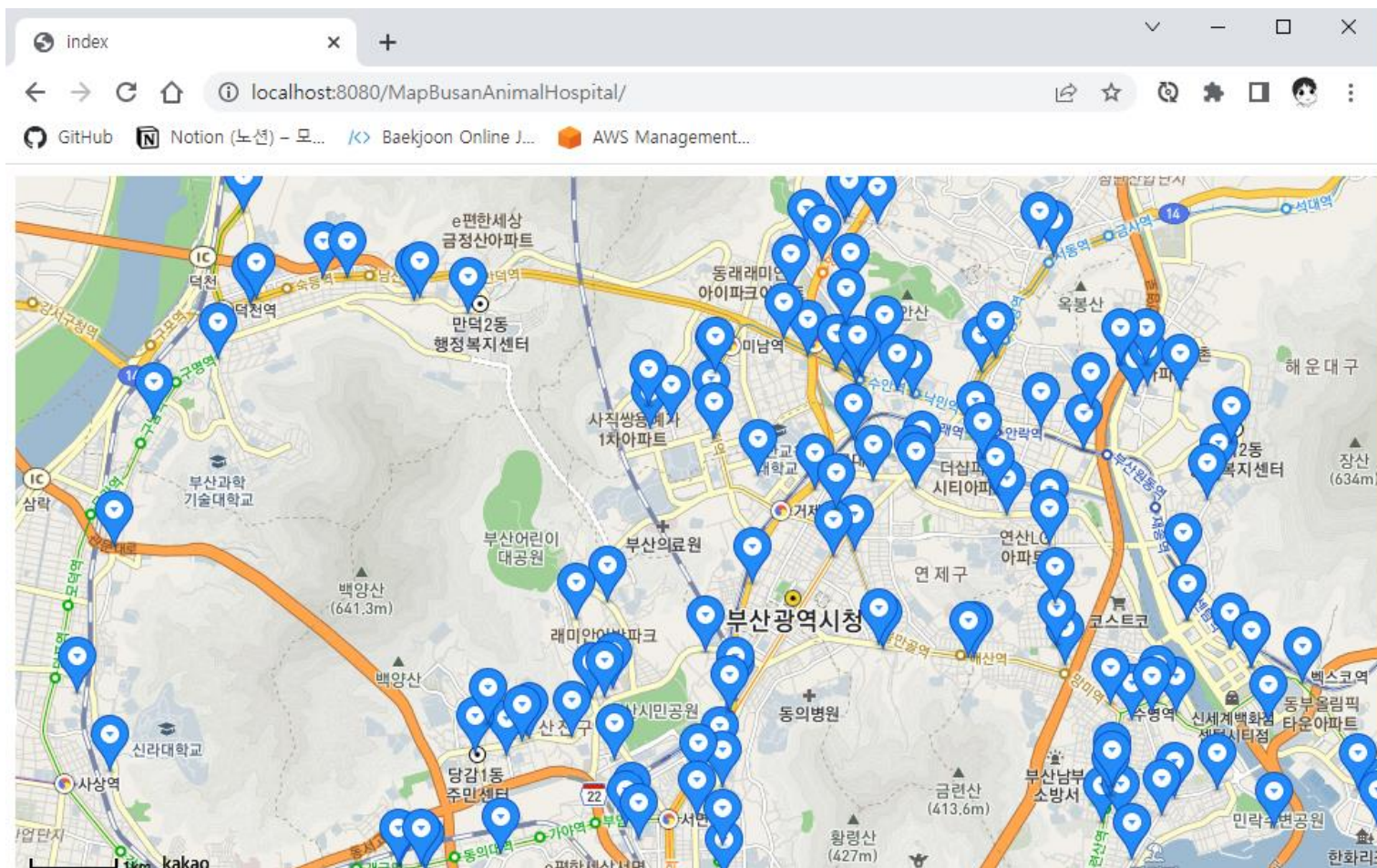
Spring Web WEB

Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.

Thymeleaf TEMPLATE ENGINES

A modern server-side Java template engine for both web and standalone environments. Allows HTML to be correctly displayed in browsers and as static prototypes.

3) 공공데이터를 활용한 서비스 개발



4) CODE

```
@Getter
@Setter
@AllArgsConstructor
@NoArgsConstructor
@Override
public class ItemVO {
    no usages
    private String gugun;
    no usages
    private String animal_hospital;
    no usages
    private String approval;
    no usages
    private String road_address;
    no usages
    private String tel;
    no usages
    private String lat;
    no usages
    private String lon;
    no usages
    private String basic_date;
}
```

```
@Getter
@Setter
@AllArgsConstructor
@NoArgsConstructor
@Override
public class ResultVO {
    no usages
    private GetTblAnimalHospital getTblAnimalHospital;

    1 usage new *
    @Getter
    @Setter
    public class GetTblAnimalHospital {
        no usages
        private Header header;
        no usages
        private Body body;

        1 usage new *
        @Getter
        @Setter
        public class Header {
            no usages
            private String resultCode;
            no usages
            private String resultMsg;
        }
    }
}
```

```
1 usage new *
@Getter
@Setter
public class Body {
    no usages
    private Items items;
    no usages
    private int numOfRows;
    no usages
    private int pageNo;
    no usages
    private int totalCount;

    1 usage new *
    @Getter
    @Setter
    public class Items {
        no usages
        private List<ItemVO> item;
    }
}
```

4) CODE

```
@Controller
public class MainController {

    no usages new *
    @GetMapping(value = {"/", "/index"})
    public String index(Model model) {
        // API 정보
        String apiURL = "http://apis.data.go.kr/62600000/BusanAn:
        String serviceKey =
        String resultType = "json";
        String pageNo = "1";
        String numOfRows = "1000"; // 부산시 전체 동물 병원 279

        URI uri = UriComponentsBuilder
            .fromUriString(apiURL)
            .queryParams(name: "serviceKey", serviceKey)
            .queryParams(name: "resultType", resultType)
            .queryParams(name: "pageNo", pageNo)
            .queryParams(name: "numOfRows", numOfRows)
            .encode()
            .build(encoded: true)
            .toUri();
```

```
RequestEntity<Void> req = RequestEntity.get(uri).build();
```

```
RestTemplate restTemplate = new RestTemplate();
```

```
ResponseEntity<String> result = restTemplate.exchange(req, String.class);
```

```
// JSON 문자열
```

```
String jsonData = result.getBody();
```

```
// JSON 파싱(Deserializtion)
```

```
ObjectMapper om = new ObjectMapper();
```

```
try {
```

```
    ResultVO resultVO = om.readValue(jsonData, ResultVO.class);
```

```
    List<ItemVO> items = resultVO.getGetTblAnimalHospital().getBody().getItems().getItem();
```

```
    //System.out.println(items);
```

```
    model.addAttribute(attributeName: "items", items);
```

```
} catch (JsonMappingException e) {
```

```
    e.printStackTrace();
```

```
} catch (JsonProcessingException e) {
```

```
    e.printStackTrace();
```

```
}
```

```
return "index";
```

4) CODE

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
  <meta charset="UTF-8">
  <title>index</title>
</head>
<body>
  <div id="map" style="position:fixed; width:100%; height:100%"></div>

  <script type="text/javascript" src="//dapi.kakao.com/v2/maps/sdk.js?appkey=
  <script th:inline="javascript">
    /**/

    var container = document.getElementById('map'); //지도를 담을 영역의 DOM 리
    var options = { //지도를 생성할 때 필요한 기본 옵션
      center: new kakao.maps.LatLng(35.17976009603529, 129.0750629182266)
      level: 5 //지도의 레벨(확대, 축소 정도)
    };

    var map = new kakao.maps.Map(container, options); //지도 생성 및 객체 리턴</pre></div><div data-bbox="529 280 977 796" data-label="Text"><pre>var items = /*[[${items}]]*/;

for(var item of items) {
  // 마커 생성
  var marker = new kakao.maps.Marker({
    title: item.animal_hospital,
    position: new kakao.maps.LatLng(item.lat,item.lon)
  });
  // 마커 표시
  marker.setMap(map);
}

/*]]&gt;*/
&lt;/script&gt;
&lt;/body&gt;
&lt;/html&gt;</pre></div>
```