

# 모던 웹 애플리케이션 개발1

2023.02.06 박진휘

# Part 1 프로젝트 생성 및 구성

**New Spring Starter Project**

⚠ A project with name 'MapBusanAnimalHospital' already exists in the workspace.

Service URL:

Name:

☒ Use default location

Location:

Type:  Packaging:

Java Version:  Language:

Group:

Artifact:

Version:

Description:

Package:

Working sets

☐ Add project to working sets

Working sets:

# Part 2 Controller, VO 구현

```
@GetMapping(value = {"/", "/index"})
public String index(Model model) {
    String apiURL = "http://apis.data.go.kr/6260000/BusanAnimalHospService/getTblAnimalHospital";
    String serviceKey = "비공개";
    String resultType = "json";
    String pageNo = "1";
    String numOfRows = "100";

    URI uri = UriComponentsBuilder.fromUriString(apiURL)
        .queryParams("serviceKey", serviceKey)
        .queryParams("resultType", resultType)
        .queryParams("pageNo", pageNo)
        .queryParams("numOfRows", numOfRows)
        .build(true).toUri();
    RequestEntity<Void> req = RequestEntity.get(uri).build();

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

    //Json
    String jsonData = result.getBody();

    //파싱
    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("items", items);
    } catch (JsonMappingException e) {
        e.printStackTrace();
    } catch (JsonProcessingException e) {
        e.printStackTrace();
    }
    return "index";
}
```

```
@Getter
@Setter
public class ResultVO {

    private GetTblAnimalHospital getTblAnimalHospital;

    @Getter
    @Setter
    public class GetTblAnimalHospital{
        private Header header;
        private Body body;

        @Getter
        @Setter
        public class Header {
            private String resultCode;
            private String resultMsg;
        }

        @Getter
        @Setter
        public class Body {
            private Items items;
            private int numOfRows;
            private int pageNo;
            private int totalCount;

            @Getter
            @Setter
            public class Items {
                private List<ItemVO> item;
            }
        }
    }
}
```

```
@Getter
@Setter
public class ItemVO {

    private String gugun;
    private String animal_hospital;
    private String approval;
    private String road_address;
    private String tel;
    private String lat;
    private String lon;
    private String basic_date;
}
```

MainController, ResultVO, ItemVO

## Part 2 HTML 구현

```
<!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=비공개"></script>
    <script th:inline="javascript">
      var container = document.getElementById('map'); //지도를 담을 영역의 DOM 레퍼런스
      var options = { //지도를 생성할 때 필요한 기본 옵션
        center: new kakao.maps.LatLng(35.17976009603529, 129.0750629182266), //지도의 중심좌표.
        level: 5 //지도의 레벨(확대, 축소 정도)
      };

      var map = new kakao.maps.Map(container, options); //지도 생성 및 객체 리턴
      var items = [{items}];

      for(var item of items){
        var marker = new kakao.maps.Marker({
          title: item.animal_hospital,
          position: new kakao.maps.LatLng(item.lat, item.lon)
        });
        marker.setMap(map);
      }
    </script>
  </body>
</html>
```

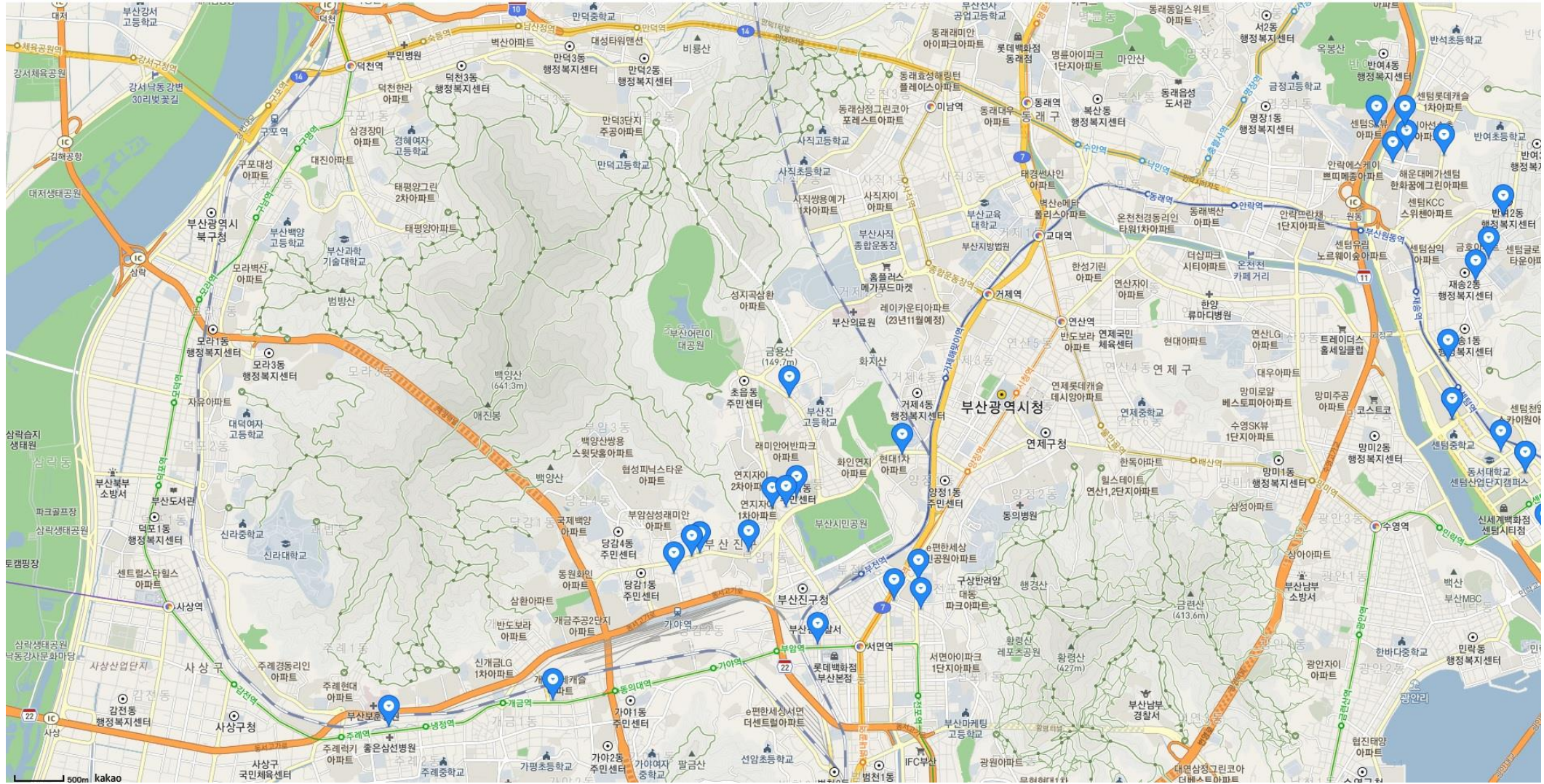
## Part 3 실행

```
{
  "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"
          },
          {
            "gugun": "해운대구",
            "animal_hospital": "갯을동물병원",
            "approval": "2004-08-16",
            "road_address": "부산광역시 해운대구 재반로 148, 32호 (재송동)",
            "tel": "051-784-7844",
            "lat": "35.190959",
            "lon": "129.127957",
            "basic_date": "2023-01-17"
          },
          {
            "gugun": "해운대구",
            "animal_hospital": "김준완 동물의료센터",
            "approval": "1999-01-22",
            "road_address": "부산광역시 해운대구 해운대로 580, 4층 (우동)",
            "tel": "051-704-7582",
            "lat": "35.161211",
            "lon": "129.155099",
            "basic_date": "2023-01-17"
          }
        ]
      }
    }
  }
}
```

Postman을 이용한 데이터 확인



# Part 3 실행



공공데이터와 카카오API를 이용한 지도 (실행 화면)