



# 프로그래밍 언어응용 문제해결 시나리오

(스마트웹 & 콘텐츠개발)자바(Java)&코틀린  
(Kotlin) 기반 스프링(Spring) 풀스택 개발  
(A)

조 만현

# 부산광역시 중요 지역 풍향 풍속 알리기



## 1. 주요 목적

- 공공 데이터 포털의 API를 이용해 데이터를 받아 프로그램 구현
- 받은 데이터를 parsing 하여 원하는 데이터를 추출



# 부산광역시 중요 지역 풍향 풍속 알리미

## 2. 선택한 API ( 부산광역시 실시간 풍향 풍속 )

### - 이유

요즘 온도에 비해 바람이 많이 불어서  
밖의 온도를 예측하기 힘든 적이 많았다.

JAVA만으로는 힘들겠지만 나중에  
JSP 나 Android 와 함께 개발을 해내고 싶고,  
더 나아가 풍향 풍속 뿐만 아닌 날씨 자체를  
커스텀하게 만들고 싶은 나의 첫 시작점이 될 것 이다.

환경기상

부산광역시

활용신청 [승인] 부산광역시\_실시간 풍향풍속 정보 서비스

계정 개발

신청일

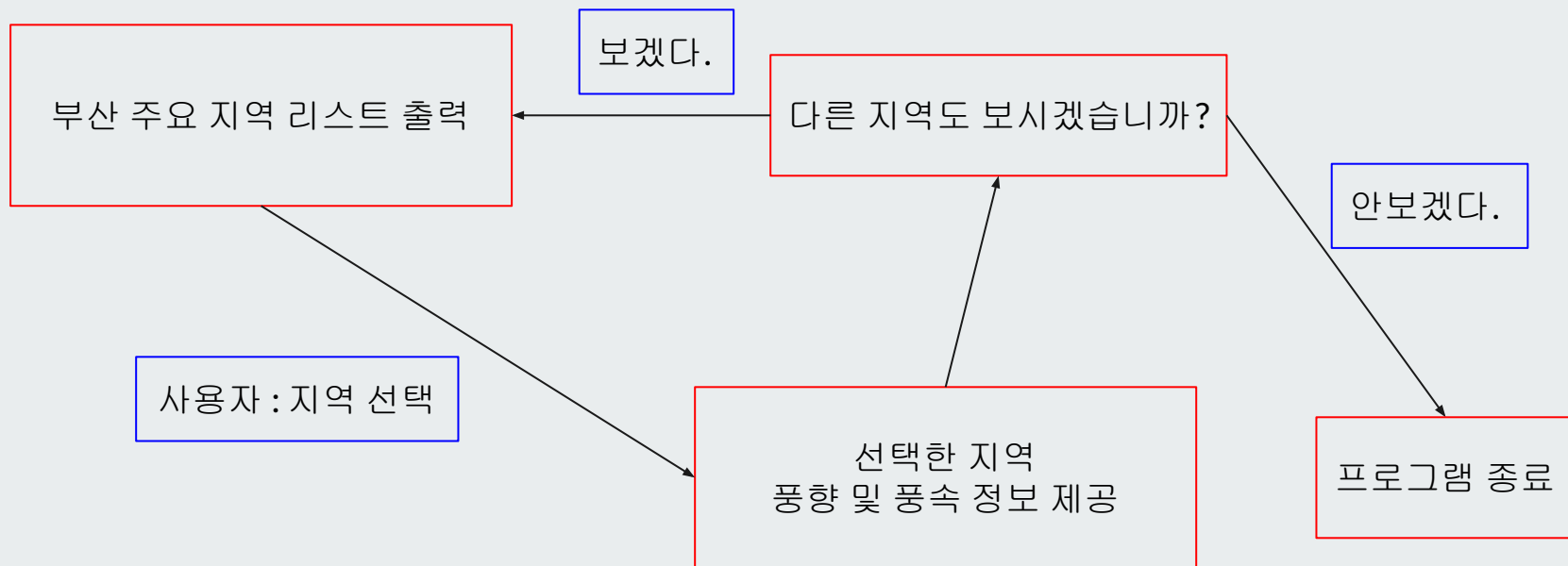
2024-03-29

만료예정일

2026-03-29

# 부산광역시 동물병원 현황

## 3. 프로그램 흐름



# 부산광역시 동물병원 현황

## 4. 주요 코드 설명

### - 클래스 ( WindSpeed )

#### - 생성자

공공 데이터 포털에서 DATA를 받아와  
필드 변수에 데이터 저장.

1. Get 형식을 사용하므로,  
요청 URL 에 파라미터와 인수를 추가
2. 요청한 URL에서 정상적인 반응이면  
rd라는 변수에 값을 담는다.
3. StringBuilder 변수에 받아온 rd 의  
데이터를 받아오며, 열려있던  
rd 와 conn은 닫는다. ( 메모리 )
4. 받아온 JSON 데이터를 최 상위 단계부터  
하나씩 받아와 원하는 ITEM 부분(data)을  
JSONArray 에 다 담는다.

```
public WindSpeed() throws IOException {
    StringBuilder urlBuilder = new StringBuilder("http://apis.data.go.kr/6260000/BusanWindInfoService/getWindInfo"); /*URL*/
    urlBuilder.append("?") + URLEncoder.encode("serviceKey", "UTF-8")
        + "=서비스키"; /*Service Key*/
    urlBuilder.append("&" + URLEncoder.encode("pageNo", "UTF-8") + "="
        + URLEncoder.encode("1", "UTF-8")); /*페이지 수*/
    urlBuilder.append("&" + URLEncoder.encode("numOfRows", "UTF-8") + "="
        + URLEncoder.encode("20", "UTF-8")); /*한 페이지당 표출 데이터 수*/
    urlBuilder.append("&" + URLEncoder.encode("resultType", "UTF-8")
        + "=" + URLEncoder.encode("json", "UTF-8")); // json 으로 출력

    URL url = new URL(urlBuilder.toString());
    HttpURLConnection conn = (HttpURLConnection) url.openConnection();
    conn.setRequestMethod("GET");
    conn.setRequestProperty("Content-type", "application/json");

    System.out.println("Response code: " + conn.getResponseCode());

    BufferedReader rd;
    if (conn.getResponseCode() >= 200 && conn.getResponseCode() <= 300) {
        rd = new BufferedReader(new InputStreamReader(conn.getInputStream()));
    } else {
        rd = new BufferedReader(new InputStreamReader(conn.getErrorStream()));
    }

    StringBuilder sb = new StringBuilder();
    String line;
    while ((line = rd.readLine()) != null) { sb.append(line); }
    rd.close();
    conn.disconnect();

    JSONTokener tokenener = new JSONTokener(sb.toString());
    JSONObject jsonString = new JSONObject(tokenener);
    JSONObject all = (JSONObject) jsonString.get("getWindInfo");
    JSONObject body = (JSONObject) all.get("body");
    JSONObject items = (JSONObject) body.get("items");
    this.item = (JSONArray) items.get("item");
}
```

# 부산광역시 동물병원 현황

## 4. 주요 코드 설명

- 클래스 ( WindSpeed )

- getArea() 메소드

- 받아온 data 중 사용자가 선택하도록  
지역 이름만 출력하여 보여주는  
메소드

- 1. 받아온 item 배열에서 지역명만  
숫자 + 지역명 으로 출

```
void getArea(){
    for(int i = 1 ; i < item.length() ; i++){
        JSONObject ob = (JSONObject)item.get(i-1);
        System.out.print((i) + "\t" + printString(ob, "siteName"));
        if(i % 4 == 0)
            System.out.println();
    }
}
```

# 부산광역시 동물병원 현황

## 4. 주요 코드 설명

### - 클래스 (App)

- selectIdx 메소드  
사용자가 보고 싶은 지역을  
정하는 메소드.

1. 사용자의 입력을 String으로 받는다.
2. 입력에 다른 값이 들어갈 수 있기에  
try/catch 문으로 예외 처리
3. 문제가 없으면 선택한 인덱스 값을  
반환한다.

```
public static int selectIdx(Scanner in, WindSpeed data){
    data.getArea();
    System.out.println();
    System.out.println("-----");
    int _idx;
    while(true) {
        System.out.print("위의 보기 중 원하는 중심 구역의 번호를 입력해주세요. : ");
        String idx = in.nextLine();
        try {
            _idx = Integer.parseInt(idx);
            if (_idx < 0 || _idx >= data.getSize()) {
                System.out.println("잘못된 입력 값 입니다.");
            } else
                return _idx;
        } catch (NumberFormatException e) {
            System.out.println("잘못된 입력 값 입니다.");
        }
    }
}
```

# 부산광역시 동물병원 현황

## 4. 주요 코드 설명

### - 클래스 ( windSpeed )

#### - 메소드 ( getItem )

사용자가 입력한 index를 인수로 받아 원하는 데이터를 제공해주는 메소드.

1. 사용자가 입력하는 idx 는 1부터 시작.
2. item변수에서 data를 받아올 때 idx 에서 1 을 빼준다.
3. 원하는 데이터에서 , JSONObject와 그 Object의 속성명을 입력하여 데이터를 받아와 출력한다.

```
void getItem(int idx) {
    JSONObject ob = (JSONObject) item.get(idx-1);
    System.out.println("-----");
    System.out.println("측정 시간\t\t" + printString(ob, "obsrDt"));
    System.out.println("지역명\t\t\t" + printString(ob, "siteName"));
    System.out.println("1m당 풍속\t\t" + printString(ob, "windSpeed1m"));
    System.out.println("1m당 풍향\t\t" + printString(ob, "windDirection1m")
        + getDerectionNm(ob, "windDirection1m"));
    System.out.println("돌풍 풍속\t\t" + printString(ob, "windSpeedGust"));
    System.out.println("돌풍 풍향\t\t" + printString(ob, "windDirectionGust")
        + getDerectionNm(ob, "windDirectionGust"));
    System.out.println("상태\t\t\t\t" + printString(ob, "sttusNm"));
    System.out.println("-----");
}
```



# 부산광역시 동물병원 현황

## 5. 실행 화면

Response code: 200

안녕하세요. 부산시 풍향풍속 알리미 입니다.

1 우동어촌계	2 대변항	3 강변대로4	4 청사포 어촌계
5 민락 어촌계	6 눌차동	7 녹산수문	8 환경공단
9 다대해수욕장	10 송도어촌계	11 감지해변	12 영도하수처리장
13 감만시민공원	14 광안리해수욕장	15 공수마을	16 두호마을
17 임랑해수욕장	18 칠암마을	19 다대항	

위의 보기 중 원하는 중심 구역의 번호를 입력해주세요. : 6

측정 시간 2024-03-29 17:04:01  
지역명 눌차동  
1m당 풍속 4.6  
1m당 풍향 306.6 북서 풍  
돌풍 풍속 170400.0  
돌풍 풍향 0.0 북 풍  
상태 정상

다른 지역도 보시겠습니까 ? ( 1. 네 / 2. 아니오 ) : 1

1 우동어촌계	2 대변항	3 강변대로4	4 청사포 어촌계
5 민락 어촌계	6 눌차동	7 녹산수문	8 환경공단
9 다대해수욕장	10 송도어촌계	11 감지해변	12 영도하수처리장
13 감만시민공원	14 광안리해수욕장	15 공수마을	16 두호마을
17 임랑해수욕장	18 칠암마을	19 다대항	

위의 보기 중 원하는 중심 구역의 번호를 입력해주세요. : 16

측정 시간 2024-03-29 17:03:08  
지역명 두호마을  
1m당 풍속 0.6  
1m당 풍향 297.7 북서 풍  
돌풍 풍속 1.3  
돌풍 풍향 284.9 서 풍  
상태 정상

다른 지역도 보시겠습니까 ? ( 1. 네 / 2. 아니오 ) : 2

부산시 풍향풍속 알리미를 이용해주셔서 감사합니다.

종료 코드 0(으)로 완료된 프로세스

# 보완점

## 1. jsonObject 출력 시 객체 내의 객체(item)를 불러와야 함 getWindInfo > body > items > item

```
JSONTokener tokenizer = new JSONTokener(sb.toString());  
JSONObject jsonString = new JSONObject(tokenizer);  
JSONObject all = (JSONObject) jsonString.get("getWindInfo");  
JSONObject body = (JSONObject) all.get("body");  
JSONObject items = (JSONObject) body.get("items");  
this.item = (JSONArray) items.get("item");
```

```
{"getWindInfo": {  
  "header": {  
    "resultCode": "00",  
    "resultMsg": "NORMAL_CODE"  
  },  
  "body": {  
    "pageNo": 1,  
    "totalCount": 20,  
    "items": {"item": [  
      {  
        "siteCode": "00-600-0029",  
        "windDirectionNmGust": null,  
        "obsrDt": "2024-03-29 17:04:04",  
        "sttusNm": "정상",  
        "siteName": "우동여촌계",  
        "windDirectionNm1m": null,  
        "windDirection10m": "",  
        "windSpeedGust": "2.8",  
        "windSpeed10m": "",  
        "windSpeed1m": "2.1",  
        "sttus": "00",  
        "windDirection1m": "246.3",  
        "windDirectionGust": "182.8",  
        "windDirectionNm10m": null  
      },  
      {  
        "siteCode": "00-600-0028",  
        "windDirectionNmGust": null,  
        "obsrDt": "2024-03-29 17:04:03",  
        "sttusNm": "정상",  
        "siteName": "대변항",  
        "windDirectionNm1m": null,  
        "windDirection10m": "",  
        "windSpeedGust": "3.1",  
        "windSpeed10m": "",  
        "windSpeed1m": "2.6",  
        "sttus": "00",  
        "windDirection1m": "90.1",  
        "windDirectionGust": "206.1",  
        "windDirectionNm10m": null  
      }  
    ]  
  }  
}
```