

생각하라! 표현하라! 코딩하라!



AVATPC
생각 표현 코딩

JavaThinkingPresentationCoding

☞ Check Point

- 클래스에 대한 정확한 이해가 필요한 사람
- 객체지향 개념의 이해가 필요한 사람
- 다형성 개념을 활용하고 싶은 사람
- 다양한 API를 활용해보고 싶은 사람
- JSON, XML Data를 핸들링 해보고 싶은 사람
- Open API를 활용해서 프로젝트를 해보고 싶은 사람

- 이해하기 쉬운 그림으로 설명
- 제작 실무교재 제공
- github 소스 제공

JavaTPC
박매일 강사

강의경력

- (현)한국전력공사 코딩 위탁 교육
 - 2019 In-House 코딩 위탁교육
 - 디지털변환 관련 기초 코딩(Python) 교육
- (현)광주소프트웨어마이스터고 산학협력교사
 - IOT실무프로젝트, 리눅스시스템프로그래밍
- 조선대학교 산업기술융합대학원 외래강사
- 한국산업인력공단 NCS기업활용 컨설턴트
- ICU 정보통신교육원, 조선대, 호남대 강의
- 리눅스마스터 공인강사, 정보처리기사
- 직업능력개발훈련교사(정보처리, 멀티미디어, 사무자동화)
- 2015'창업경진대회 중소기업청장상(우수상)
- 2016,17' 한전KDN 에너지SW경진대회수상

PART -3

목차

기존강의 : JavaTPC(생각하고, 표현하고, 코딩하고) - PART-1, 2

Java TPC 실전프로젝트(Java API 활용)

Project01 : Java Naver Maps OpenAPI 활용 Geocoding(지도서비스) 프로젝트

Project02 : Java Jsoup API 활용 Crawling(크롤링) 프로젝트

Project03 : Java Naver Search OpenAPI 활용 Excel 프로젝트

Project04 : Java iText API 활용 PDF 프로젝트

Project05 : Java MQTT Client 만들기(온.습도 모니터링 및 LED제어)

Project06 : Java Socket(소켓) Multi-Chatting 프로그래밍

강의자료 및 소스파일 주소

<https://github.com/bitcocom/JavaTPCProject>

JavaTPC(PART3).pdf - 강의파일

1. 객체를 표현해보자

2명의 회원정보를 표현하려고 한다. 어떻게 표현하는 것이 가장 좋을까요? 문자열의 형태로 표현해보자

ex) 회원은 : 이름, 주소, 전화번호를 가지고 있는 객체입니다.

1) 의미 없고 구분이 있는 문자열 형태

```
String textMember= "홍길동,광주,010-1111-1111#나길동,서울,010-2222-2222" ;
```

단순하지만 데이터 처리의 어려움

2) 의미 있고 구분 있는 문자열 형태(XML)

```
String xmlMember= "<members>" +
    "<member>" +
        "<name>홍길동</name>" +
        "<address>광주</address>" +
        "<phone>010-1111-1111</phone>" +
    "</member>" +
    "<member>" +
        "<name>나길동</name>" +
        "<address>서울</address>" +
        "<phone>010-2222-2222</phone>" +
    "</member>" +
"</members>" ;
```

데이터처리는 쉬우나 데이터 크기가 커짐

API

3) 의미 있고 구분 있는 문자열 형태(JSON)

```
String jsonMember=" [ { 'name' : '홍길동' , 'address' : '광주' , 'phone' : '010-1111-1111' } , " +
    " { 'name' : '나길동' , 'address' : '서울' , 'phone' : '010-2222-2222' } ] " ;
```

데이터 크기는 작아지나 표현이 제한적임

API

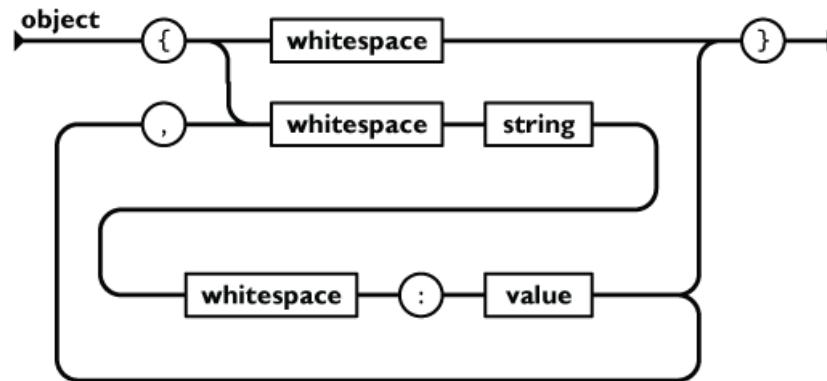
Gson / JSON-Java

2. JSON 이란? <https://www.json.org/>

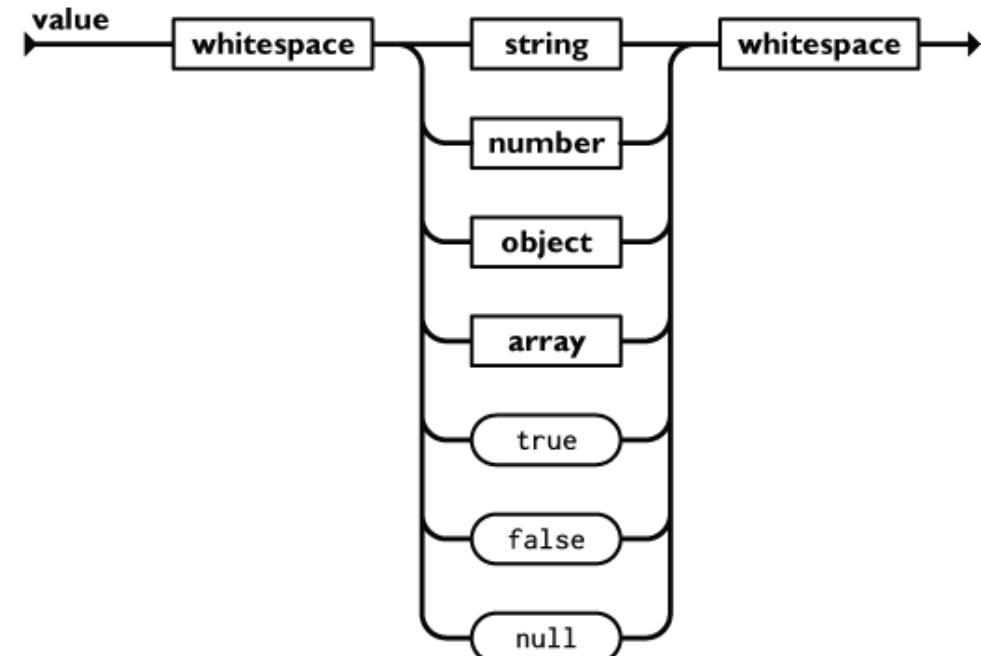
JSON (JavaScript Object Notation) is a lightweight data-interchange format

플랫폼이나 언어와 무관하게 자료 교환을 목적으로 만들어진 포맷이다

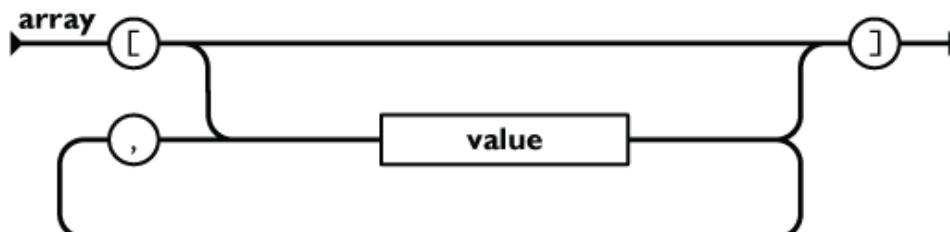
An **object** is an unordered set of name/value pairs. An object begins with $\{_{left\ brace}$ and ends with $\}_{right\ brace}$. Each name is followed by $:_{colon}$ and the name/value pairs are separated by $,_{comma}$.



A **value** can be a *string* in double quotes, or a *number*, or **true** or **false** or **null**, or an **object** or an **array**. These structures can be nested.



An **array** is an ordered collection of values. An array begins with $[_{left\ bracket}$ and ends with $]_{right\ bracket}$. Values are separated by $,_{comma}$.



3. Gson API 사용하기

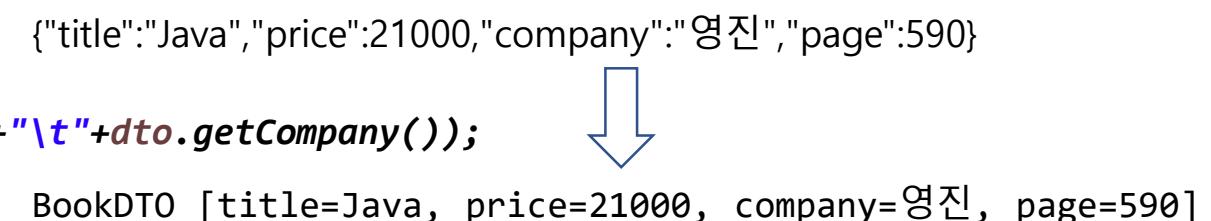
1. Gson을 활용하여 Object를 JSON으로 바꾸기

```
BookDTO vo=new BookDTO("Java",21000,"영진",590);
Gson g=new Gson();
String json=g.toJson(vo);
System.out.println(json);
```



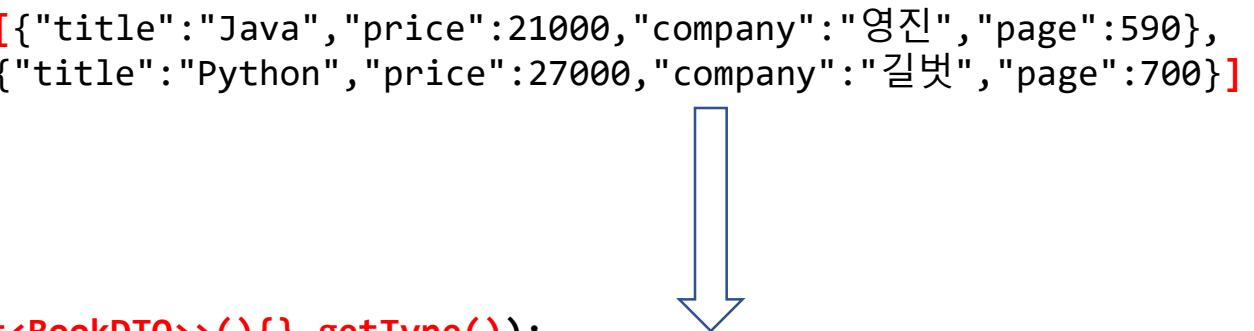
2. Gson을 활용하여 JSON을 Object로 바꾸기

```
BookDTO dto=g.fromJson(json, BookDTO.class);
System.out.println(dto.toString());
System.out.println(dto.getTitle()+"\t"+dto.getPrice()+"\t"+dto.getCompany());
```



3. Gson을 활용하여 List를 JSON으로 바꾸기

```
List<BookDTO> list=new ArrayList<BookDTO>();
list .add(new BookDTO("Java",21000,"영진",590));
list .add(new BookDTO("Python",27000,"길벗",700));
String gList=g.toJson(list);
System.out.println(gList);
```

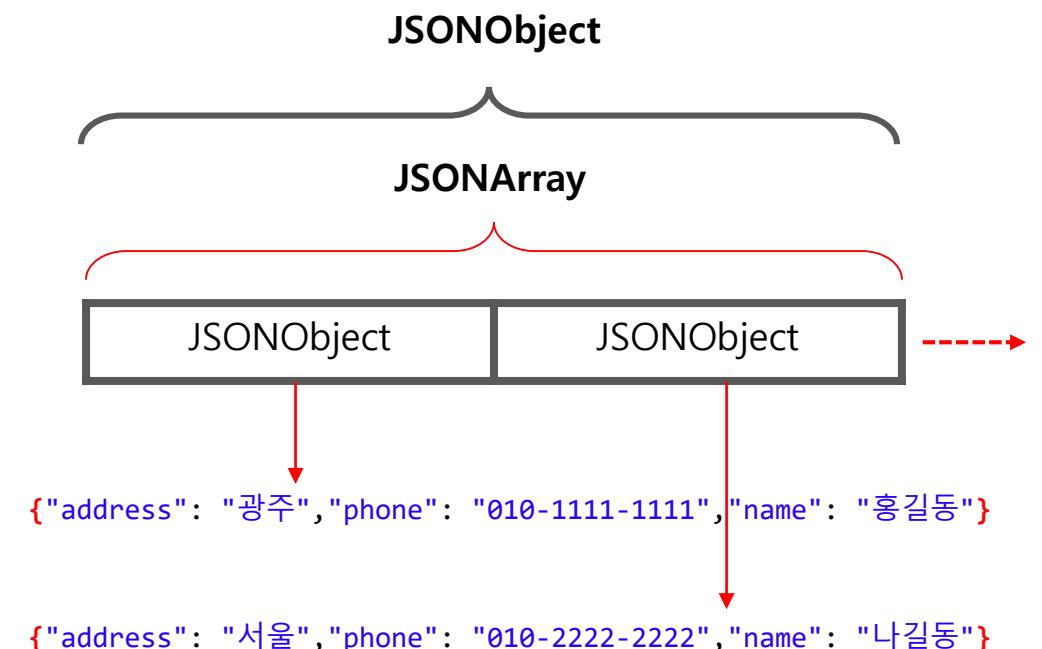
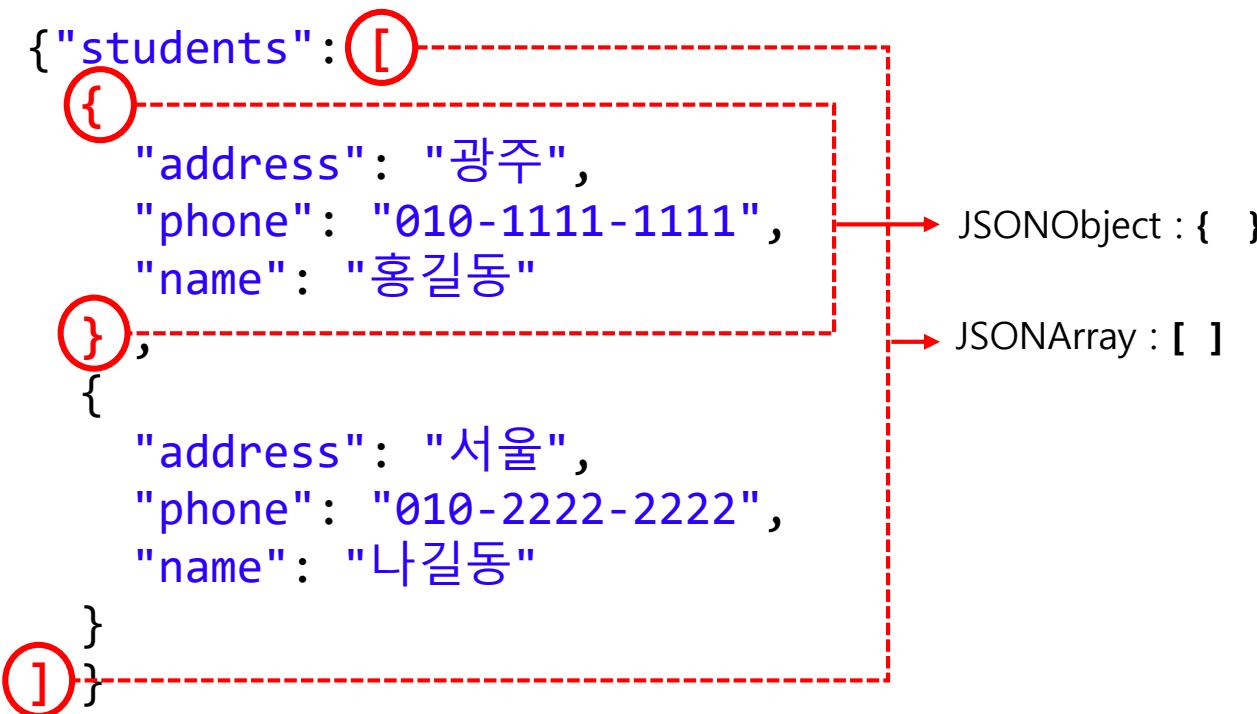


4. Gson을 활용하여 JSON을 List로 바꾸기

```
List<BookDTO> lst=g.fromJson(gList, new TypeToken<List<BookDTO>>(){}.getType());
for(BookDTO v : lst) {
    System.out.println(v);
}
```

BookDTO [title=Java, price=21000, company=영진, page=590]
BookDTO [title=Python, price=27000, company=길벗, page=700]
BookDTO [title=Oracle, price=30000, company=대림, page=890]

1. JSON-Java(org.json)



```
JSONObject student = new JSONObject();
student.put("name", "홍길동");
student.put("address", "광주");
student.put("phone", "010-1111-1111");
```



```
System.out.print(" - " + student.get("name"));
System.out.print(" - " + student.get("address"));
System.out.println(" - " + student.get("phone"));
```

2. JSON Data 분석 : <https://jsoneditoronline.org>

The screenshot shows the JSON Editor Online interface. On the left, there is a code editor window with the following JSON code:

```
1 {"students": [2 {3 "address": "광주",4 "phone": "010-1111-1111",5 "name": "홍길동"},6 {7 "address": "서울",8 "phone": "010-2222-2222",9 "name": "나길동"},10 ]}11121314
```

The first two student objects are highlighted with a red dashed box. On the right, there is a tree view of the same JSON structure. The path 'object > students > 1' is selected. The tree shows the following structure:

- object {1}
 - students [2]
 - 0 {3}
 - address : 광주
 - phone : 010-1111-1111
 - name : 홍길동
 - 1 {3}
 - address : 서울
 - phone : 010-2222-2222
 - name : 나길동

PROJECT 1
주소를 입력하면 위도와 경도를 추출하는 프로그래밍

PART-3

1. 네이버 클라우드 플랫폼에 회원가입 및 로그인하기

→회원 가입은 어떻게 하나요?

<https://www.ncloud.com/support/faq/member/3784>

The screenshot shows the NAVER CLOUD PLATFORM website with a red dashed box around the URL bar containing 'https://www.ncloud.com/'. A red arrow points from the question '→회원 가입은 어떻게 하나요?' to the '회원가입' link in the top right navigation bar. The page content discusses the membership application process, mentioning steps like selecting country and member type, agreeing to terms of service, entering member information, and adding payment methods.

네이버 클라우드 플랫폼의 회원가입하는 절차는 아래와 같습니다.

(거주지) 국가 및 회원 유형 선택 -> 약관동의 -> 회원정보입력 -> 결제수단등록
 * 회원정보입력이 완료되면 회원가입도 완료되지만, 원활한 서비스 이용을 위해서는 결제수단을 등록이 필요합니다.

1) 국가 및 회원 유형 선택
 회원 가입하려는 사용자의 거주지 국가를 선택하고, 회원 유형을 개인과 사업자 중에서 선택합니다.
</support/downloadAttachedFile?bbsFileNo=1587&articleNo=1114&lang=ko>

2) 약관 동의
 서비스 이용약관, 개인정보수집이용에 대한 안내 문구를 확인하고 동의합니다.

3) 회원정보입력

- 개인: 미성년자 확인을 위해 생년월일을 입력해야 합니다. 만 19세 미만의 사용자는 가입하실 수 없습니다.
- 사업자: '한국'이 거주지인 사업자는 사업자등록번호, 대표자명, 업종/업태 정보를 입력해야 하며, 사업자등록증 사본을 '첨부파일로 제출' 항목을 선택하여 제출해주세요 합니다.

 (별도로 제출하실 경우 '고객지원 문의하기로 별도 제출' 항목을 선택하여 고객지원 > 문의하기를 통해 파일 첨부하여 제출해주시면 됩니다.)

4) 결제수단등록

- 회원정보입력을 완료하면 회원가입완료와 결제수단등록 안내가 함께 노출됩니다. 원활한 서비스 이용을 위해서는 결제수단등록이 필요합니다.
- 해외 사용자와, '한국'이 거주지인 개인은 신용카드로만 결제가 가능합니다.(한국이 거주지인 사업자는 신용카드와 계좌이체 등을 선택할 수 있습니다.)

2. 네이버 클라우드 플랫폼의 지도 API 서비스 등록 방법

1. 네이버 클라우드 플랫폼 포털(www.ncloud.com)에 회원 가입을 합니다.

* 참고: 회원가입을 어떻게 하나요? 바로가기

→네이버 클라우드 플랫폼의 지도 API 서비스 등록하는 방법은 아래와 같습니다.
<https://www.ncloud.com/support/faq/product/3511>

2. 로그인 후 결제수단을 등록합니다. 등록 후 콘솔에 접속합니다.

(콘솔 접속 버튼은 홈페이지 메인화면에서 우측 상단에 위치)

*참고: 결제수단 등록하는 절차는 어떻게 되나요? 바로가기

*참고: 요금납부 기간은 어떻게 되나요? 바로가기

3. 콘솔에 접속 후, [All product > AI-Application Service > AI/ NAVER API > Maps] 상품을 선택합니다.

4. [+Application등록] 버튼을 눌러, 서비스 등록을 시작합니다.

5. 약관 동의 후, [App 이름 설정], [Service 선택], [서비스 환경 등록]을 순차적으로 진행합니다.

서비스할 Application의 이름을 설정해주신 후, 사용하실 API명에 체크해주시면 됩니다.

※ 단, 신규 추가된 Tile Map, Directions, Search Places 3개의 API는 이관 프로모션 혜택이 적용되지 않으니, 이 점 유의하셔서 선택해주시기 바랍니다.

※ 2019년 1월 1일부터 12월 31일까지 일정량 무료 프로모션을 진행하고 있습니다. (Maps API 요금 및 프로모션과 관련된 상세 내용은 상품 소개 페이지를 참고해주세요.)

* 참고: 상품 소개 바로가기

서비스 환경 등록은 서비스 하실 URL을 필수로 등록해주세요 하며 최대 10개까지 등록 가능합니다.

안드로이드 앱 패키지 이름과 iOS번들 ID도 등록 가능하며, 등록하실 경우 정확하게 입력해주세요 합니다.

6. 서비스 환경 등록까지 마치면 서비스 신청이 완료되며, 등록된 Application을 확인할 수 있고, 설정한 Application 이름 아래에 [인증정보]라고 쓰인 열쇠모양 아이콘이 있습니다.

Client ID를 복사한 후 Tutorials > 시작하기 > Hello, World의 NAVER 지도 API v3 로드하기의 ncpClientId=YOUR_CLIENT_ID에 대신 넣어서 사용하시면 됩니다.

* 참고: Tutorials > 시작하기 > Hello, World의 NAVER 지도 API v3 로드하기 바로가기

3. 네이버 클라우드 플랫폼의 지도 API 서비스 등록

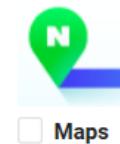
The screenshot shows the Naver Cloud Platform console interface. The top navigation bar includes 'Console' (highlighted with a red dashed box), a back arrow, a refresh icon, and a URL field showing <https://console.ncloud.com/mc/solution/naverService/application>. The main content area has two tabs: 'Kubernetes Service' (highlighted with a green button) and 'Update'. On the left sidebar, 'Region 한국 / KR 한국어' is selected, and 'All Products' is highlighted with a yellow dashed box. The main panel displays 'Storage' services: Object Storage, File Storage, and NAS. To the right, under 'AI-NAVER API', several services are listed: Clova Speech Recognition (CSR), Clova Speech Synthesis (CSS), Clova Face Recognition (CFR), Papago NMT (highlighted with a green button), Papago Korean Name Remanizer, and Maps (highlighted with a green button). Below this, the 'Application' section is shown with a red dashed box around its header. It contains a message: '등록한 Application 정보를 확인하고 관리합니다.' (Check and manage registered Application information). A blue button '+ Application 등록' is at the bottom left of this section. At the very bottom, there is a table with columns: App 이름 (App Name), 서비스구분 (Service Category), 당일 사용량 (Daily Usage), and 당월 사용량 (Monthly Usage). The table lists three services: JavaTPC (Static Map, Search Places, Geocoding), all with 0% usage.

App 이름	서비스구분	당일 사용량	당월 사용량
JavaTPC	Static Map Search Places Geocoding	0%	0/300,000 회 0/75,000 회 0/300,000 회

4. 네이버 클라우드 플랫폼의 지도 API 서비스 등록

AI-NAVER API / Application

Application ①



Maps

Application 이름 설정

AI · Naver Service 를 이용할 Application 을 등록하세요

Application 이름

JavaTPC

- Web Dynamic Map
- Mobile Dynamic Map
- Static Map
- Tile Map
- Directions 5
- Directions 15
- Search Places
- Geocoding
- Reverse Geocoding

서비스 환경 등록

Application에서 이용할 서비스 환경을 등록하세요

Web 서비스 URL(최대 10개)

http://tpc.kr

+추가

우측의 [추가] 버튼을 클릭해야 Web 서비스 URL 입력이 완료됩니다.

Android 앱 패키지 이름(최대 10개)

kr.tpc

+추가

우측의 [추가] 버튼을 클릭해야 Android 앱 패키지 입력이 완료됩니다.

iOS Bundle ID(최대 10개)

iOS앱의 Bundle ID를 입력하세요. 예) com.naver.example.MyNaverMap

+추가

취소

등록

서비스 설명/요금안내	개발 가이드

AI-NAVER API / Application

Application ②

등록한 Application 정보를 확인하세요

인증 정보

```
con.setRequestProperty("X-NCP-APIGW-API-KEY-ID", clientId);
con.setRequestProperty("X-NCP-APIGW-API-KEY", clientSecret);
```

Application 이름 JavaTPC

Client ID (X-NCP-APIGW-API-KEY-ID)

Client Secret (X-NCP-APIGW-API-KEY)

서비스환경

Web 서비스 URL http://tpc.kr

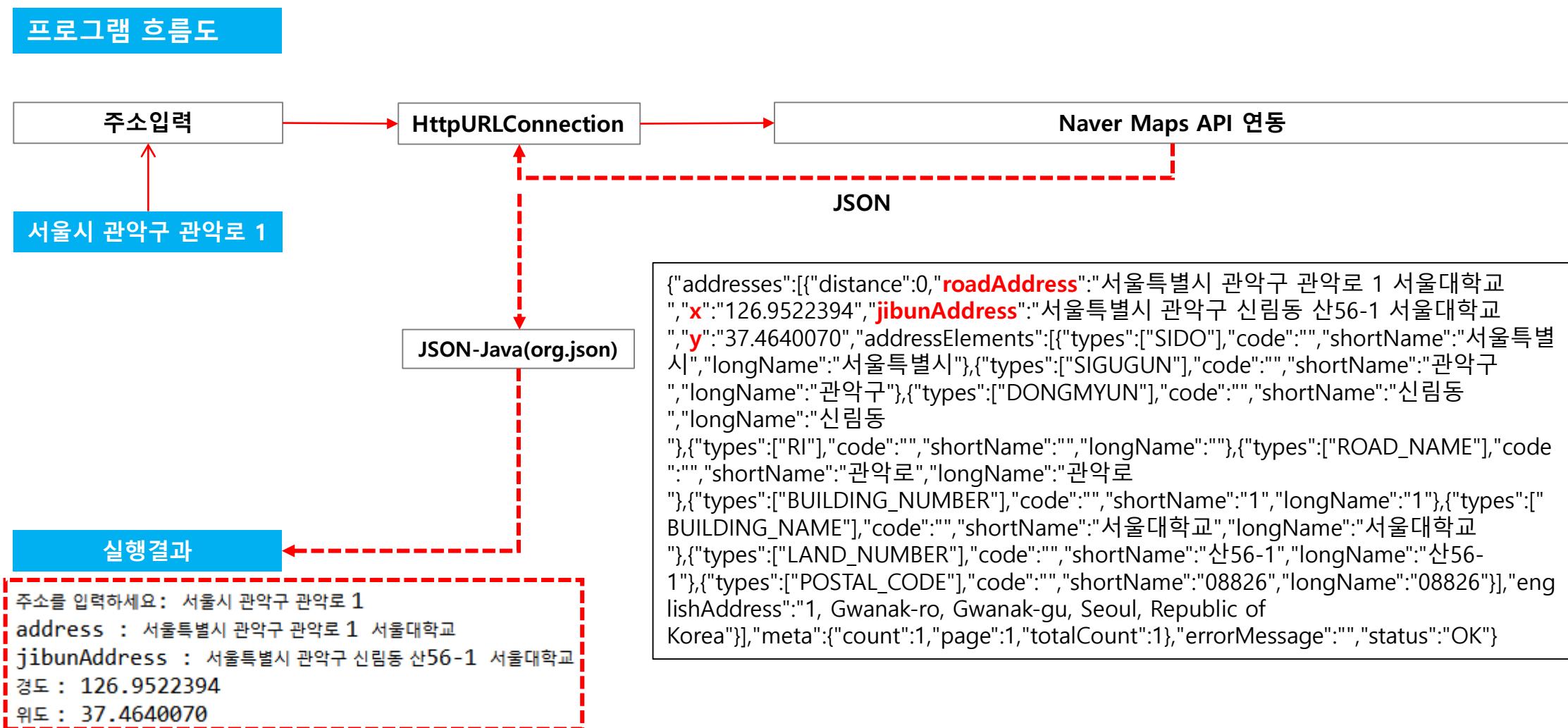
Android 앱 패키지 이름 kr.tpc

iOS Bundle ID

정보 변경을 원하시면, Application 선택 후 변경 버튼을 통해 진행해주세요.

✓ 확인

1. 주소를 입력하면 위도와 경도를 추출하는 프로그래밍 – Console Version



2. JSON Data 분석 : <https://jsoneditoronline.org/>

The screenshot displays the JSON Editor Online interface. The left pane shows the raw JSON code for a geocoding request:

```
1 {"addresses": [{"distance": 0, "roadAddress": "서울특별시 관악구 관악로 1 서울대학교", "x": "126.9522394", "jibunAddress": "서울특별시 관악구 신림동 산56-1 서울대학교", "y": "37.4640070", "addressElements": [{"types": ["SIDO"], "code": "", "shortName": "서울특별시", "longName": "서울특별시"}, {"types": ["SIGUNG"], "code": "", "shortName": "관악구", "longName": "관악구"}, {"types": ["DONGMYUN"], "code": "", "shortName": "신림동", "longName": "신림동"}, {"types": ["RI"], "code": "", "shortName": "", "longName": ""}, {"types": ["ROAD_NAME"], "code": "", "shortName": "관악로", "longName": "관악로"}, {"types": ["BUILDING_NUMBER"], "code": "", "shortName": "1", "longName": "1"}, {"types": ["BUILDING_NAME"], "code": "", "shortName": "서울대학교", "longName": "서울대학교"}, {"types": ["LAND_NUMBER"], "code": "", "shortName": "산56-1", "longName": "산56-1"}, {"types": ["POSTAL_CODE"], "code": "", "shortName": "08826", "longName": "08826"}], "englishAddress": "1, Gwanak-ro, Gwanak-gu, Seoul, Republic of Korea"}, "meta": {"count": 1, "page": 1, "totalCount": 1}, "errorMessage": "", "status": "OK"}]
```

The right pane shows the JSON structure with expanded address elements. The address elements are highlighted in red boxes:

- 도로명 주소: roadAddress : 서울특별시 관악구 관악로 1 서울대학교
- 경도: x : 126.9522394
- 지번: jibunAddress : 서울특별시 관악구 신림동 산56-1 서울대학교
- 위도: y : 37.4640070

Below these, the addressElements array is expanded, showing the full address details.

3. Geocoding OpenAPI 주소



- HOME**
- > API 가이드
- > Compute
- > Storage
- > Database
- > Networking
- > Security
- > Management
- AI·NAVER**
 - > Clova Speech Recognition(CSR)
 - > Clova Speech Synthesis(CSS)
 - > Clova Face Recognition(CFR)
 - > Object Detection
 - > Pose Estimation
 - > Papago NMT
 - > Papago Language Detection
 - > Papago Korean Name Romanizer
 - > Static Map
 - > Tile Map
 - > Directions 5
 - > Directions 15
 - > Search Places
 - > Geocoding**

API URL

메서드	인증	요청 URL	응답 형식
GET	ID,KEY	https://naveropenapi.apigw.ntruss.com/map-geocode/v2/geocode	JSON

요청 헤더

헤더 명	설명
X-NCP-APIGW-API-KEY-ID	앱 등록 시 발급받은 Client ID <code>X-NCP-APIGW-API-KEY-ID:{Client ID}</code>
X-NCP-APIGW-API-KEY	앱 등록 시 발급 받은 Client Secret <code>X-NCP-APIGW-API-KEY:{Client Secret}</code>

오퍼레이션

`curl "https://naveropenapi.apigw.ntruss.com/map-geocode/v2/geocode?query={주소}&coordinate=#{검색_중심_좌표}" \`

• **geocode**

```

-H "X-NCP-APIGW-API-KEY-ID: {애플리케이션 등록 시 발급받은 client id값}" \
-H "X-NCP-APIGW-API-KEY: {애플리케이션 등록 시 발급받은 client secret값}" -v

```

주소 검색 API는 지번, 도로명을 질의어로 사용해서 주소 정보를 검색합니다.
검색 결과로 주소 목록과 세부 정보를 JSON 형태로 반환합니다.

4. Source Code_1

```

import java.io.*;
import java.net.*;
import java.util.Date;
import org.json.JSONArray;
import org.json.JSONObject;
import org.json.JSONTokener;
public class Project01_D {
public static void main(String[] args) {
    // 주소 -> 위도, 경도 좌표 얻어오기
    BufferedReader io = new BufferedReader(new InputStreamReader(System.in));
    String clientId = "네이버 지도 API 키";
    String clientSecret = "네이버 지도 API 비밀번호";
    try {
        System.out.print("주소를 입력하세요: ");
        String address = io.readLine();
        String addr = URLEncoder.encode(address, "UTF-8"); // 입력 공백도 문자처리 해줘야 한다.
        String apiURL = "https://naveropenapi.apigw.ntruss.com/map-geocode/v2/geocode?query=" + addr; //json
        URL url = new URL(apiURL);
        HttpURLConnection con = (HttpURLConnection)url.openConnection();
        con.setRequestMethod("GET");
        con.setRequestProperty("X-NCP-APIGW-API-KEY-ID", clientId);
        con.setRequestProperty("X-NCP-APIGW-API-KEY", clientSecret);

        int responseCode = con.getResponseCode();
    }
}

```

주소를 입력하세요: 서울시 관악구 관악로 1
address : 서울특별시 관악구 관악로 1 서울대학교
jibunAddress : 서울특별시 관악구 신림동 산56-1 서울대학교
경도 : 126.9522394
위도 : 37.4640070



5. Source Code_2

```

BufferedReader br;
if(responseCode==200) { // 정상 호출 (200)
    br = new BufferedReader(new InputStreamReader(con.getInputStream(), "UTF-8"));
} else {
    br = new BufferedReader(new InputStreamReader(con.getErrorStream()));
}
String inputLine;
StringBuffer response = new StringBuffer(); //문자열 추가 변경시 사용
while ((inputLine = br.readLine()) != null) {
    response.append(inputLine);
}
br.close();

```

JSON-Java(org.json)

```

JSONTokener tokener=new JSONTokener(response.toString());
JSONObject object=new JSONObject(tokener);
//System.out.println(object);

JSONArray arr=object.getJSONArray("addresses");
for(int i=0;i<arr.length();i++){
    JSONObject temp = (JSONObject) arr.get(i);
    System.out.println("address : "+temp.get("roadAddress"));
    System.out.println("jibunAddress : "+temp.get("jibunAddress"));
    System.out.println("경도 : "+temp.get("x"));
    System.out.println("위도 : "+temp.get("y"));
}
} catch (Exception e) {
    System.out.println(e);
}
}

```

map_service(경도, 위도, 주소); // 지도 이미지 생성 메서드 호출
 x, y, roadAddress



2. JSON.simple

com.googlecode.json-simple ➤ json-simple

A simple Java toolkit for JSON

```

JSONParser jpr = new JSONParser();
JSONObject jarr = (JSONObject) jpr.parse(response.toString());
JSONArray arr = (JSONArray)jarr.get("addresses");
//System.out.println(object);
for(int i=0;i<arr.size();i++){
    JSONObject temp = (JSONObject) arr.get(i);
    System.out.println("address : "+temp.get("roadAddress"));
    System.out.println("jibunAddress : "+temp.get("jibunAddress"));
    System.out.println("경도 : "+temp.get("x"));
    System.out.println("위도 : "+temp.get("y"));
}

```

1. Static Map API 주소

AI·NAVER

- > Clova Speech Recognition(CSR)
- > Clova Speech Synthesis(CSS)
- > Clova Face Recognition(CFR)
- > Object Detection
- > Pose Estimation
- > Papago NMT
- > Papago Language Detection
- > Papago Korean Name Romanizer

Static Map

- raster

- > Tile Map
- > Directions 5
- > Directions 15
- > Search Places
- Geocoding**

 - geocode

- > Reverse Geocoding
- > nShortURL

API URL

메서드	요청 URI	응답 형식	인증 방식
GET	https://naveropenapi.apigw.ntruss.com/map-static/v2/raster	Binary(PNG/JPG 이미지 형식)	ID-KEY 인증
GET	https://naveropenapi.apigw.ntruss.com/map-static/v2/raster-cors	Binary(PNG/JPG 이미지 형식)	HTTP Referer 인증

요청 헤더

헤더명	설명
X-NCP-APIGW-API-KEY-ID	앱 등록 시 발급받은 Client ID X-NCP-APIGW-API-KEY-ID:{Client ID}
X-NCP-APIGW-API-KEY	앱 등록 시 발급 받은 Client Secret X-NCP-APIGW-API-KEY:{Client Secret}

오퍼레이션

- raster

```
curl "https://naveropenapi.apigw.ntruss.com/map-static/v2/raster?w=300&h=300&center=127.1054221,37.3591614&level=16" \
-H "X-NCP-APIGW-API-KEY-ID: {애플리케이션 등록 시 발급받은 client id값}" \
-H "X-NCP-APIGW-API-KEY: {애플리케이션 등록 시 발급받은 client secret값}" -v
```

2. 위도와 경도를 추출 후 지도를 이미지로 저장하기 - Console Version

```

map_service(x,y,z);
public static void map_service(String point_x, String point_y, String address) {
    String URL_STATICMAP = "https://naveropenapi.apigw.ntruss.com/map-static/v2/raster?";
    try {
        String pos=URLEncoder.encode(point_x + " " + point_y, "UTF-8");
        String url = URL_STATICMAP;
        url += "center=" + point_x + "," + point_y;
        url += "&level=16&w=700&h=500";
        url += "&markers=type:t|size:mid|pos:"+pos+"|label:" +URLEncoder.encode(address, "UTF-8");
        URL u = new URL(url);
        HttpURLConnection con = (HttpURLConnection)u.openConnection();
        con.setRequestMethod("GET");
        con.setRequestProperty("X-NCP-APIGW-API-KEY-ID", "?");
        con.setRequestProperty("X-NCP-APIGW-API-KEY", "?");
        int responseCode = con.getResponseCode();
        BufferedReader br;
        if(responseCode==200) { // 정상 호출
            InputStream is = con.getInputStream();
            int read = 0;
            byte[] bytes = new byte[1024];
            // 랜덤한 이름으로 파일 생성
            String tempname = Long.valueOf(new Date().getTime()).toString();
            File f = new File(tempname + ".jpg");
            f.createNewFile();
            OutputStream outputStream = new FileOutputStream(f);
            while ((read = is.read(bytes)) != -1) {
                outputStream.write(bytes, 0, read);
            }
            is.close();
        } else { // 예외 발생
            br = new BufferedReader(new InputStreamReader(con.getErrorStream()));
            String inputLine;
            StringBuffer response = new StringBuffer();
            while ((inputLine = br.readLine()) != null) {
                response.append(inputLine);
            }
            br.close();
            System.out.println(response.toString());
        }
    } catch (Exception e) {
        System.out.println(e);
    }
}

```

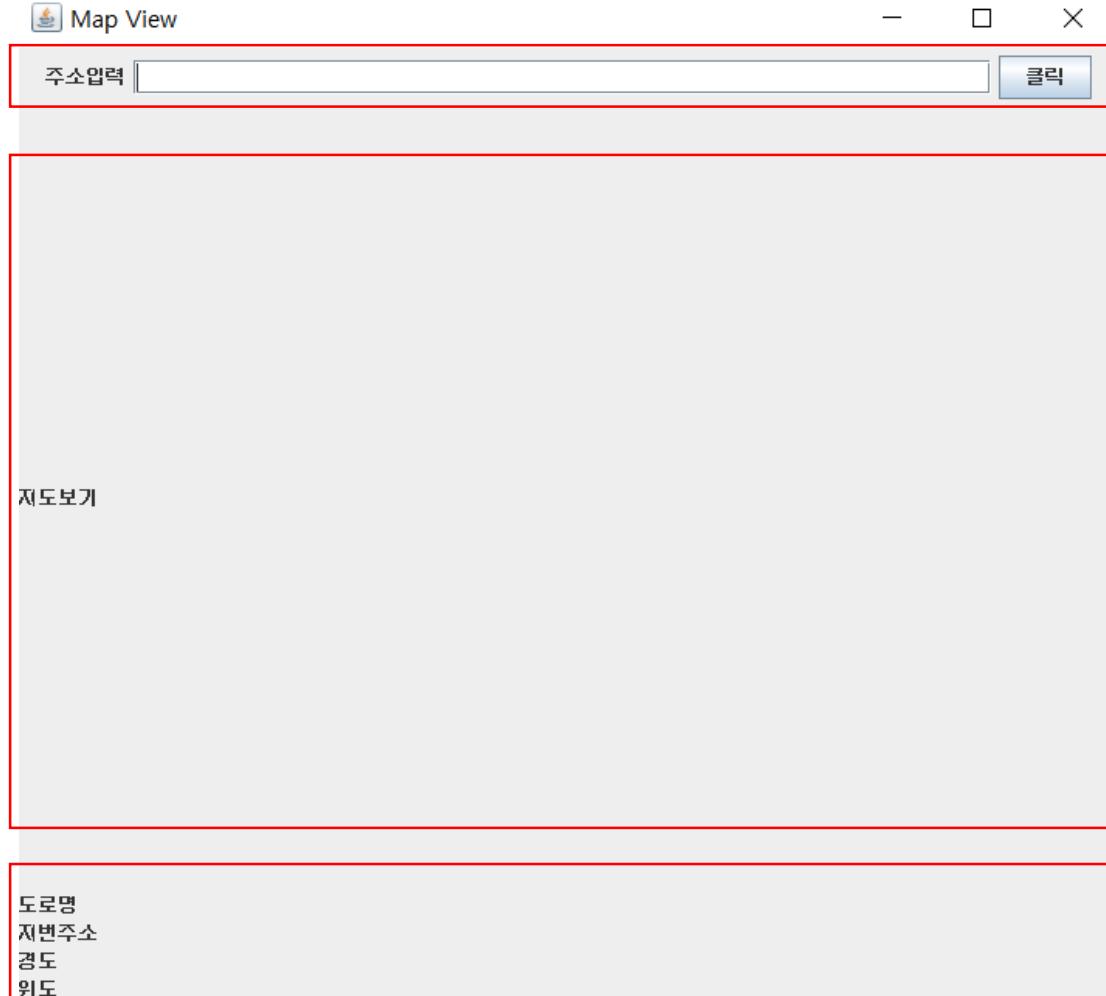
The code above is a Java console application for generating a map image from coordinates and an address. It uses the Naver Map Static API. The `map_service` method takes three parameters: `point_x`, `point_y`, and `address`. It constructs a URL with these parameters and sends a GET request to the API. If the response code is 200 (success), it reads the response stream into a buffer and writes it to a file named after the current timestamp with a .jpg extension. If there's an error, it reads the error stream and prints the response to the console.

Java Project Structure:

- JavaTPCProject
- src
- JRE System Library
- Referenced Libraries
 - 1570611181893.jpg
 - 1570611253715.jpg

Map Image:

1. 위도와 경도를 추출하여 지도를 보여주는 프로그래밍 – GUI Version



2. GUI 만들기– Source Code_1(Project01_F.java)

```

import javax.swing.*;
import java.awt.*;
public class Project01_F{
JTextField address;
JLabel resAddress, resX, resY, jibunAddress, imageLabel;
public void initGUI() {
    JFrame frm=new JFrame("Map View");
    frm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    Container c = frm.getContentPane();
    imageLabel = new JLabel("지도보기");
    JPanel pan=new JPanel();
    JLabel addressLbl = new JLabel("주소입력");
    address=new JTextField(50);
    JButton btn=new JButton("클릭");
    pan.add(addressLbl);
    pan.add(address);
    pan.add(btn);
    btn.addActionListener(new NaverMap(this));
    JPanel pan1=new JPanel();
    pan1.setLayout(new GridLayout(4, 1));
    resAddress = new JLabel("도로명");
    jibunAddress = new JLabel("지번주소");
    resX = new JLabel("경도");
    resY = new JLabel("위도");
    pan1.add(resAddress);
    pan1.add(jibunAddress);
    pan1.add(resX);
    pan1.add(resY);
}
}

```

```

c.add(BorderLayout.NORTH, pan);
c.add(BorderLayout.CENTER, imageLabel);
c.add(BorderLayout.SOUTH, pan1);
frm.setSize(730,660);
frm.setVisible(true);
}

public static void main(String[] args) {
    new Project01_F().initGUI();
}
}

```



3. GUI 만들기– Source Code_3(NaverMap.java)

```

    map_service(vo); ←

public void map_service(AddressVO vo) {
String URL_STATICMAP = "https://naveropenapi.apigw.ntruss.com/map-static/v2/raster?";
try {
    String pos=URLEncoder.encode(vo.getX() + " " + vo.getY(), "UTF-8");
    URL_STATICMAP += "center=" + vo.getX() + "," + vo.getY();
    URL_STATICMAP += "&level=16&w=700&h=500";
    URL_STATICMAP += "&markers=type:t|size:mid|pos:" + pos + "|label:" + URLEncoder.encode(vo.getRoadAddress(), "UTF-8");
    URL url = new URL(URL_STATICMAP);
    HttpURLConnection con = (HttpURLConnection)url.openConnection();
    con.setRequestMethod("GET");
    con.setRequestProperty("X-NCP-APIGW-API-KEY-ID", "?");
    con.setRequestProperty("X-NCP-APIGW-API-KEY", "?");
    int responseCode = con.getResponseCode();
    BufferedReader br;
    if(responseCode==200) { // 정상 호출
        InputStream is = con.getInputStream();
        int read = 0;
        byte[] bytes = new byte[1024];
        String tempname = Long.valueOf(new Date().getTime()).toString();
        File f = new File(tempname + ".jpg");
        f.createNewFile();
        OutputStream outputStream = new FileOutputStream(f);
        while ((read =is.read(bytes)) != -1) {
            outputStream.write(bytes, 0, read);
        }
        is.close();
    }
}

```

```

JSONArray arr=object.getJSONArray("addresses");
for(int i=0;i<arr.length();i++){
    JSONObject temp = (JSONObject) arr.get(i);
    // AddressVO 객체에 주소 정보를 저장
    vo=new AddressVO();
    vo.setRoadAddress((String)temp.get("roadAddress"));
    vo.setJibunAddress((String)temp.get("jibunAddress"));
    vo.setX((String)temp.get("x"));
    vo.setY((String)temp.get("y"));
    System.out.println(vo);
}
map_service(vo);

```

```

ImageIcon img=new ImageIcon(f.getName());
naverMap.imageLabel.setIcon(img);
naverMap.resAddress.setText(vo.getRoadAddress());
naverMap.jibunAddress.setText(vo.getJibunAddress());
naverMap.resX.setText(vo.getX());
naverMap.resY.setText(vo.getY());
} else { // 예외 발생
    System.out.println(responseCode);
}
} catch (Exception e) {
    System.out.println(e);
}
}

```



4. GUI 만들기– Source Code_2(AddressVO.java)

```

package kr.inflearn;
public class AddressVO {
    private String roadAddress;
    private String jibunAddress;
    private String x;
    private String y;
    public AddressVO() { }
    public AddressVO(String roadAddress, String jibunAddress, String x, String y) {
        super();
        this.roadAddress = roadAddress;
        this.jibunAddress = jibunAddress;
        this.x = x;
        this.y = y;
    }
    public String getRoadAddress() {
        return roadAddress;
    }
    public void setRoadAddress(String roadAddress) {
        this.roadAddress = roadAddress;
    }
    public String getJibunAddress() {
        return jibunAddress;
    }
    public void setJibunAddress(String jibunAddress) {
        this.jibunAddress = jibunAddress;
    }
    public String getX() {
        return x;
    }
    public void setX(String x) {
        this.x = x;
    }
    public String getY() {
        return y;
    }
    public void setY(String y) {
        this.y = y;
    }
    @Override
    public String toString() {
        return "AddressVO [roadAddress=" + roadAddress + ", jibunAddress=" +
               jibunAddress + ", x=" + x + ", y=" + y
               + "]";
    }
}

```

AddressVO

PROJECT 2 Jsoup API를 이용하여 웹 페이지 Crawling 하기

1. Jsoup API를 이용하여 웹 페이지 Crawling 하기

<https://jsoup.org>

jsoup: Java HTML Parser

jsoup is a Java library for working with real-world HTML.

It provides a very convenient API for extracting and manipulating data, using the best of DOM, CSS, and jquery-like methods.

Jsoup의 주요 요소는 크게 다섯 가지로 볼 수 있다

클래스명	설명
Document	Jsoup 얻어온 결과 HTML 전체 문서
Element	Document의 HTML 요소
Elements	Element가 모인 자료형. for나 while 등 반복문 사용이 가능하다.
Connection	Jsoup의 connect 혹은 설정 메소드들을 이용해 만들어지는 객체, 연결을 하기 위한 정보를 담고 있다.
Response	Jsoup가 URL에 접속해 얻어온 결과. Document와 다르게 status 코드, status 메시지나 charset같은 헤더 메시지와 쿠키등을 가지고 있다.

2. Jsoup를 이용해서 네이버 스포츠 크롤링(<https://sports.news.naver.com/wfootball/index.nhn>)

The screenshot shows a browser window with the Naver Sports homepage (<https://sports.news.naver.com/wfootball/index.nhn>). The page displays a news card with a headline about Messi and a list of upcoming football matches. To the right, the Chrome DevTools Elements tab is open, showing the HTML structure of the news card. A red dashed box highlights the URL in the browser's address bar, another red dashed box highlights the news card on the page, and a third red dashed box highlights the 'home_news' class in the Elements panel.

Elements Panel Screenshot:

```

<div class="home_news" style="display: flex; align-items: center; justify-content: space-between; padding: 10px; margin-bottom: 10px;">
  <div class="head" style="flex: 1; text-align: center;">
    <div class="title" style="font-size: 1.2em; font-weight: bold;">[News Title]
```

3. Jsoup를 이용해서 네이버 스포츠 크롤링

```
import java.io.IOException;
import org.jsoup.Jsoup;
import org.jsoup.nodes.*;
import org.jsoup.select.Elements;
public class Project02_A {
public static void main(String[] args) {

    String url = "https://sports.news.naver.com/wfootball/index.nhn";
    Document doc = null;
    try {
        doc = Jsoup.connect(url).get();
    } catch (IOException e) {
        e.printStackTrace();
    }

    // 주요 뉴스로 나오는 태그를 찾아서 가져오도록 한다.
    Elements element = doc.select("div.home_news");

    // 1. 헤더 부분의 제목을 가져온다.
    String title = element.select("h2").text().substring(0, 4);

    System.out.println("=====");
    System.out.println(title);
    System.out.println("=====");

    for(Element el : element.select("li")) {      // 하위 뉴스 기사들을 for문 돌면서 출력
        System.out.println(el.text());
    }
    System.out.println("=====");

}
}
```



```
<div class="home_news">
  <div class="head">
    <h2></h2>
    <a href="/wfootball/news/index.nhn" onclick="clickcr(this, 'mnl.more', '', '', event); class="more">...</a>
  </div>
  <ul class="home_news_list">...</ul>
  <ul class="home_news_list division">
    <li class>
      <a href="/wfootball/news/read.nhn?oid=139&aid=0002120492" target onclick="clickcr(this, 'mnl.txt', '880000D4_00000000000000002120492', '', event); title="토트넘, 모리뉴 택해야 하는 이유...·빅6 이길 필사적인 자'">
        <span>
          토트넘, 모리뉴 택해야 하는 이유...·빅6 이길 필사적인 자
        </span>
      </a>
    </li>
    <li class>...</li>
    <li class>...</li>
    <li class>...</li>
    <li class>...</li>
    <li class="division">...</li>
    <li class>...</li>
    <li class>...</li>
    <li class>...</li>
    <li class>...</li>
  </ul>
  =====
  주요뉴스
  =====
  판다이크 뚫고 주가상승...활희찬, 아스널 관심 받는다
  메시 징계로 빠진 아르헨티나, 독일 원정 평가전서 2-2 무
  단호한 메시, "이적 없다. 바르셀로나에서 은퇴하겠다"
  최하위 삼프도리아, 새 사령탑 후보로 라니에리에 접근(伊 언론)
  마르티네스의 믿음, "아자르 살 쪄도 차이 만들어, 쿠르투아 세계 최고 GK"
  메시, 바르사 비선실세설 부인... "조언만 할 뿐"
  바이에른-사네 측, 지난 주 만남... 이적설 재점화
  베르바토프의 일침 "베일, 존경 받지 못하는 것 이상하다"
  독일-아르헨티나 2-2 무승부... 양 팀 모두 A매치 무패 이어가
  리옹 사령탑 거절한 무리뉴, 목표는 EPL 복귀
  '1143억 이적료', 매과이어, "나에 대한 평가는 5~6년 후"
  =====
  토트넘, 모리뉴 택해야 하는 이유...·빅6 이길 필사적인 자
  =====
  위기의 맨유가 1월 싸게 살 수 있는 알짜 선수 6명, 에릭센 만주기치 등등
  감독 교체설에 주전 이적설, 'UCL 준우승' 토트넘 흔들린다
  우스만 멜벌레, 2경기 징계...엘 클라시코 결장 [오피셜]
  매과이어의 각오 "5~6년 안에 성공할 거로 믿는다"
  9일 만에 생각 바뀐 막타 '맨유, 4위 진입 쉽지 않다'
  '구단도 나선다' 팰리스, 자하 향한 인종차별에 형사 조치 예고
  '부드러워진' 케이로스?, "하메스 A매치 제외는 내 결정"
  아르헨티나의 교체카드, 테어슈테겐 뚫었다
  =====
```

PART-3

1. Jsoup API를 이용하여 웹 페이지 Crawling 하기

The screenshot illustrates a digital workflow for a sermon summary and its corresponding audio recording.

Left Panel: A sermon summary titled "오직 여호와께로 가까이" (Only to God Alone) is displayed. The summary includes a date (2019.10.10), a title (여호수아(Joshua) 23:1 - 23:16 찬송가 215장), and a list of 11 numbered points. A red box highlights the title and date.

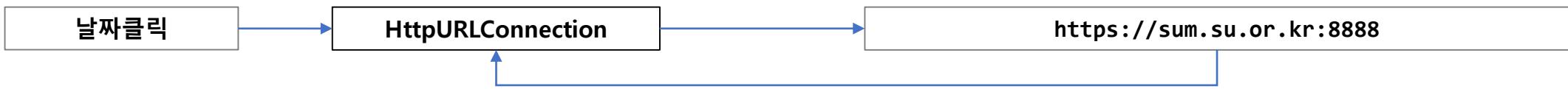
Middle Panel: A file browser interface shows the file path: > SRC > JRE System Library [JavaSE-11] > Referenced Libraries > 20191011.mp3. A red box highlights the file name "20191011.mp3".

Right Panel: An audio player interface titled "오디오해설" (Audio Commentary) displays a video thumbnail of a park scene with the text "매일성경과 함께" (Daily Bible with Us). Below the thumbnail is a progress bar showing 0:00 / 13:00. A red box highlights the file path "http://meditation.su.or.kr/meditation_mp3/2019/20191011.mp3".

Bottom Panel: A large yellow arrow points from the sermon summary on the left to the audio player on the right, indicating the connection between the text and the audio recording.

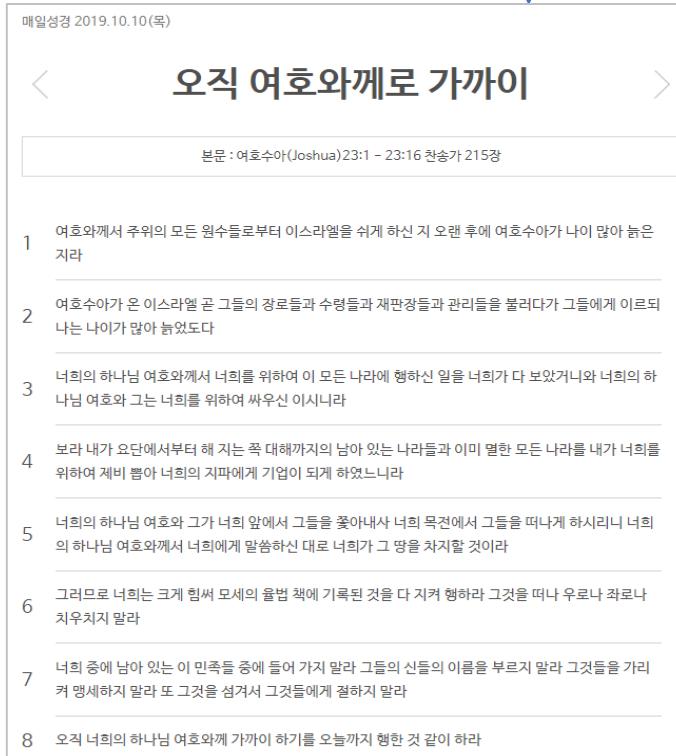
2. 웹 페이지 Crawling 하기(HTMP Parsing)

프로그램 흐름도



String bible=br.readLine();

[https://sum.su.or.kr:8888/bible/today/Ajax/Bible/BodyMatter?qt_ty=QT1&Base_de="+bible+"&bibleType=1](https://sum.su.or.kr:8888/bible/today/Ajax/Bible/BodyMatter?qt_ty=QT1&Base_de=)



jsoup API

실행결과

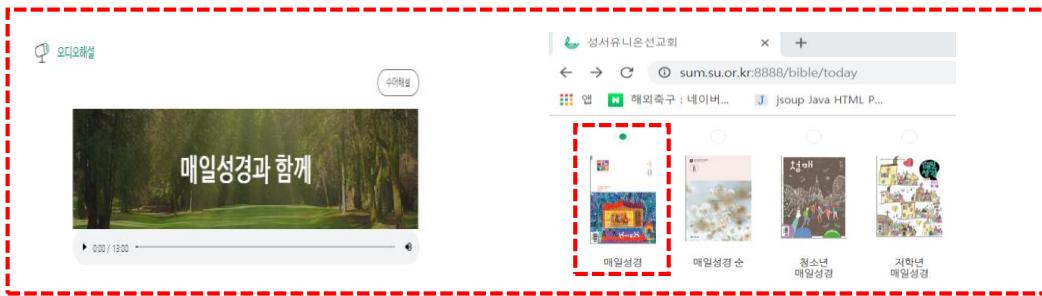
```

Document doc=Jsoup.connect(url).post();
Element bible_text=doc.select(".bible_text").first();
Element bibleinfo_box=doc.select("#bibleinfo_box").first();
Element dailybible_info=doc.select("#dailybible_info").first();
Elements liList=doc.select(".body_list > li");
for(Element li: liList) {
    System.out.print(li.select(".num").first().text()+":");
    System.out.println(li.select(".info").first().text());
}
  
```

[입력 ->년(yyyy)-월(mm)-일(dd)]:2019-10-10

실행 결과 출력:
 1:여호와께서 주위의 모든 원수들로부터 이스라엘을 쉬게 하신 지 오랜 후에 여호수아가 나이 많아 늙은지라
 2:여호수아가 온 이스라엘 곧 그들의 장로들과 수령들과 재판장들과 관리들을 불러다가 그들에게 이르되 나는 나이가 많아 늙었도다
 3:너희의 하나님 여호와께서 너희를 위하여 이 모든 나라에 행하신 일을 너희가 다 보았거니와 너희의 하나님 여호와 그는 너희를 위하여 싸우신 이시니라
 4:보라 내가 요단에서부터 해 지는 쪽 대해까지의 남아 있는 나라들과 이미 멸한 모든 나라를 내가 너희를 위하여 제비 뿐만 너희의 지
 5:너희의 하나님 여호와 그가 너희 앞에서 그들을 쫓아내사 너희 목전에서 그들을 떠나게 하시리니 너희의 하나님 여호와께서 너희에게 말씀하신 대로 너희가 그 땅을 차지할 것이라
 6:그러므로 너희는 크게 힘써 모세의 율법 책에 기록된 것을 다 지켜 행하라 그것을 떠나 우로나 좌로나 치우치지 말라
 7:너희 중에 남아 있는 이 민족들 중에 들어 가지 말라 그들의 신들의 이름을 부르지 말라 그것들을 가리켜 맹세하지 말라 또 그것을 삼
 8:오직 너희의 하나님 여호와께 가까이 하기를 오늘까지 행한 것 같이 하라
 9:이는 여호와께서 강대한 나라들을 너희의 앞에서 쫓아내셨으므로 오늘까지 너희에게 맞선 자가 하나도 없었느니라
 10:너희 중 한 사람이 천 명을 쫓으리니 이는 너희의 하나님 여호와 그가 너희에게 말씀하신 것 같이 너희를 위하여 싸우실이라
 11:그러므로 스스로 조심하여 너희의 하나님 여호와를 사랑하라

3. 웹 페이지 Crawling 하기(리소스 다운로드)



http://meditation.su.or.kr/meditation_mp3/2019/20191011.mp3

mp3

```
Element source=doc.select("source").first();
String audio=source.attr("src").trim();
System.out.println(audio);
String fileName=audio.substring(audio.lastIndexOf("/") + 1);
```

```
Runnable r=new DownloadBroker(audio, fileName);
Thread dLoad=new Thread(r);
dLoad.start();
```

```
for(int i=0;i<10;i++) {
    try {
```

```
        Thread.sleep(1000);
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

```
System.out.print(""+(i+1));
}
System.out.println();
System.out.println("=====");
```

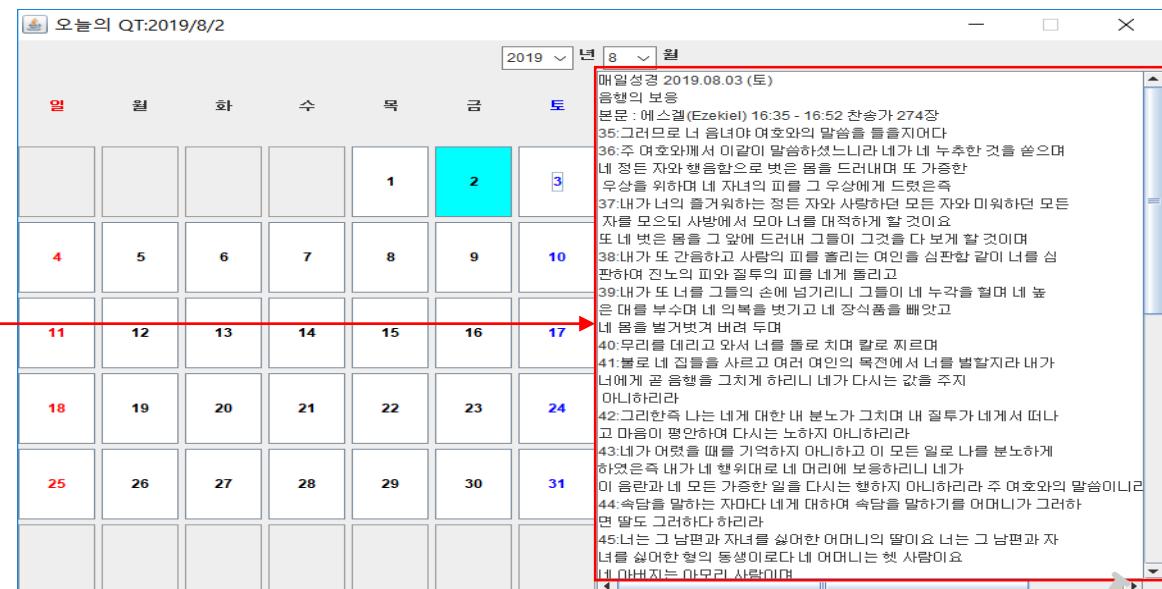
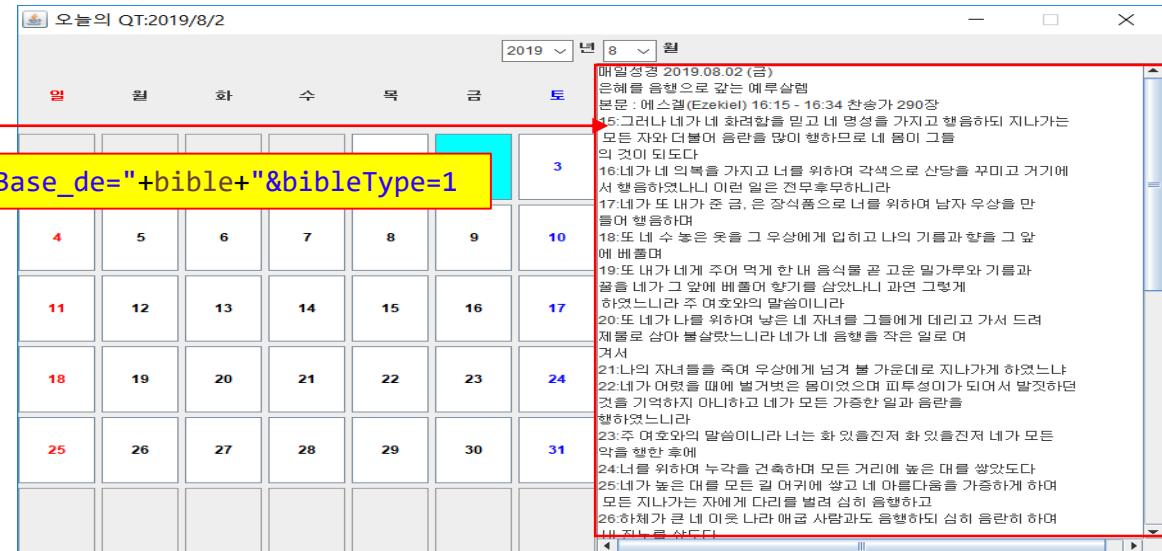
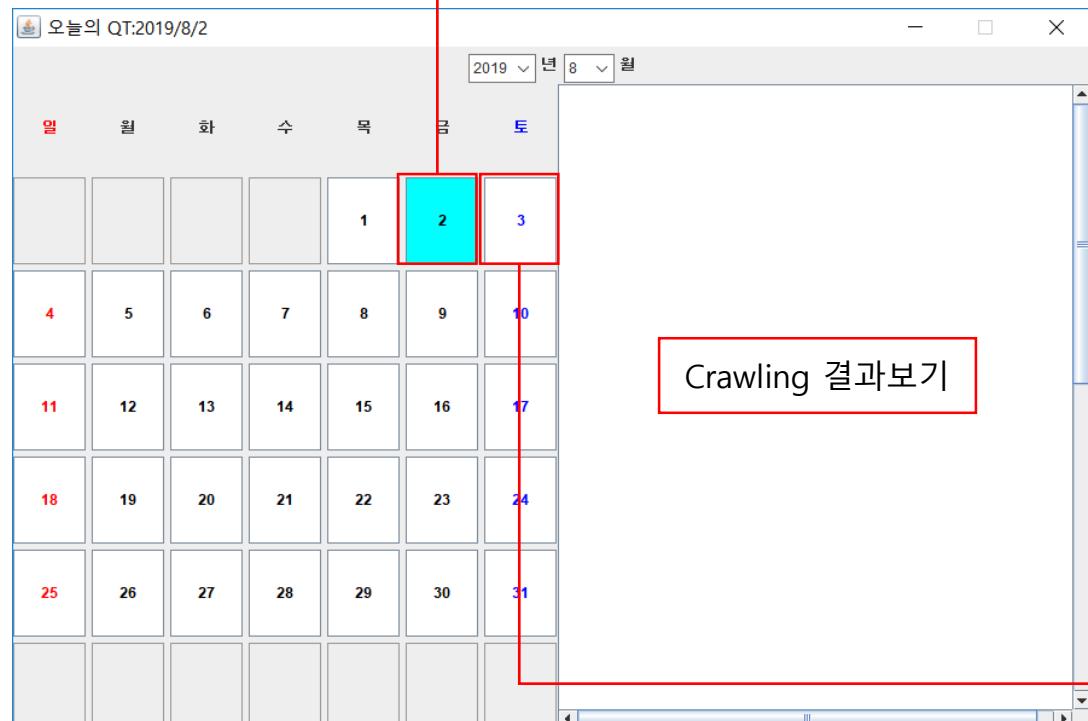
image

```
Element source=doc.select(".img > img").first();
String imgUrl="https://sum.su.or.kr:8888"+source.attr("src").trim();
System.out.println(imgUrl);
String fileName=imgUrl.substring(imgUrl.lastIndexOf("/") + 1);
Runnable r=new DownloadBroker(imgUrl, fileName);
```

```
package kr.inflearn;
import java.io.*;
import java.net.URL;
public class DownloadBroker implements Runnable {
private String address;
private String fileName;
public DownloadBroker(String address) {
    this.address=address;
}
public DownloadBroker(String address, String fileName) {
    this.address=address;
    this.fileName=fileName;
}
@Override
public void run() {
try {
    FileOutputStream fos=new FileOutputStream(fileName);
    BufferedOutputStream bos=new BufferedOutputStream(fos);
    URL url=new URL(address);
    InputStream is=url.openStream();
    BufferedInputStream input=new BufferedInputStream(is);
    int data;
    while((data=input.read())!=-1) {
        bos.write(data);
    }
    bos.close();
    input.close();
    System.out.println("download complete...");
    System.out.println(fileName);
} catch (Exception e) {
    e.printStackTrace();
}
}
}
```

1. Jsoup API를 이용하여 웹 페이지 Crawling 하기(GUI)

[https://sum.su.or.kr:8888/bible/today/Ajax/Bible/BodyMatter?qt_ty=QT1&Base_de="+bible+"&bibleType=1](https://sum.su.or.kr:8888/bible/today/Ajax/Bible/BodyMatter?qt_ty=QT1&Base_de=)



1. Source Code

```

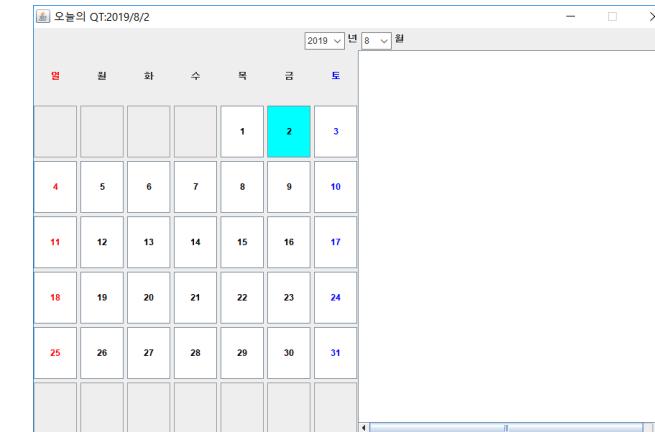
public class Project02_F extends JFrame implements ActionListener, ItemListener{
    private Choice chyear, chmonth;
    private JLabel yLabel, mLabel;
    private JTextArea area;
    GregorianCalendar gc;
    private int year, month;
    private JLabel[] dayLabel = new JLabel[7];
    private String[] day={"일","월","화","수","목","금","토"};
    private JButton[] days = new JButton[42];//7일이 6주이므로 42개의 버튼필요
    private JPanel selectPanel = new JPanel();
    private GridLayout grid = new GridLayout(7,7,5,5);//행, 열, 수평갭, 수직갭
    private Calendar ca = Calendar.getInstance();
    private Dimension dimen, dimen1;
    private int xpos, ypos;
    public Project02_F(){
        setTitle("오늘의 QT:"+ca.get(Calendar.YEAR)+"/"+(ca.get(Calendar.MONTH)+1)+"/"+ca.get(Calendar.DATE));
        setSize(900,600);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        dimen = Toolkit.getDefaultToolkit().getScreenSize();
        dimen1 = this.getSize();
        xpos = (int)(dimen.getWidth()/2 - dimen1.getWidth()/2);
        ypos = (int)(dimen.getHeight()/2 - dimen1.getHeight()/2);
        setLocation(xpos, ypos);//화면의 가운데에 출력
        setResizable(false);
        setVisible(true);
        chyear = new Choice(); chmonth = new Choice();
        yLabel = new JLabel("년"); mLabel = new JLabel("월");
        init();
    }
}

```

```

import java.awt.*;
import java.awt.event.*;
import java.util.*;
import javax.swing.*;
import org.jsoup.Jsoup;
import org.jsoup.nodes.*;
import org.jsoup.select.Elements;

```



2. Source Code

```
@Override  
public void actionPerformed(ActionEvent arg0) {  
    area.setText("");  
    String year = chyear.getSelectedItem();  
    String month = chmonth.getSelectedItem();  
    JButton btn = (JButton)arg0.getSource();  
    String day = btn.getText();  
    System.out.println(year+"."+month+"."+day);  
    String bible=year+"-"+month+"-"+day;  
    // Jsoup API : HTML파싱 방법  
    //String url="https://sum.su.or.kr:8888/bible/today";  
    String url="https://sum.su.or.kr:8888/bible/today/Ajax/Bible/BodyMatter?qt_ty=QT1&Base_de="+bible+"&bibleType=1&SearchTxt1=CalDer";  
    try {  
        Document doc=Jsoup.connect(url).post();  
        //System.out.println(doc.toString());  
        Element bible_text=doc.select(".bible_text").first();  
        System.out.println(bible_text.text());  
        Element bibleinfo_box=doc.select("#bibleinfo_box").first();  
        System.out.println(bibleinfo_box.text());  
        // dailybible_info  
        Element dailybible_info=doc.select("#dailybible_info").first();  
        System.out.println(dailybible_info.text());  
        area.append(dailybible_info.text()+"\n");  
        area.append(bible_text.text()+"\n");  
        area.append(bibleinfo_box.text()+"\n");  
        Elements liList=doc.select(".body_list > li");  
    }  
}
```

3. Source Code

```
for(Element li: liList) {
    String line=li.select(".info").first().text();
    if(line.length()>65) {
        line=line.substring(0,36)+"\n"+line.substring(36,66)+"\n"+line.substring(66)+"\n";
        area.append(li.select(".num").first().text()+": "+line);
    }else if(line.length()>35) {
        line=line.substring(0,36)+"\n"+line.substring(36)+"\n";
        area.append(li.select(".num").first().text()+": "+line);
    }else {
        area.append(li.select(".num").first().text()+": "+li.select(".info").first().text()+"\n");
    }
    System.out.print(li.select(".num").first().text()+": ");
    System.out.println(li.select(".info").first().text());
}
} catch (Exception e) {
    e.printStackTrace();
}
}

public void init(){
    select();
    calendar();
}
```

4. Source Code

```
public void select(){
    JPanel panel = new JPanel(grid); // 7행 7열의 그리드레이아웃
    for(int i=2020; i>=2000;i--){
        chyear.add(String.valueOf(i));
    }
    for(int i=1; i <=12; i++){
        chmonth.add(String.valueOf(i));
    }
    for(int i = 0; i < day.length;i++){//요일 이름을 레이블에 출력
        dayLabel[i] = new JLabel(day[i],JLabel.CENTER);
        panel.add(dayLabel[i]);
        dayLabel[i].setBackground(Color.GRAY); // 사실상 의미가 없음. 바꿔지 않음.
    }
    dayLabel[6].setForeground(Color.BLUE); // 토요일 색상
    dayLabel[0].setForeground(Color.RED); // 일요일 색상
    for(int i = 0; i < 42;i++){//모두 42개의 버튼을 생성
        days[i] = new JButton(""); // 제목이 없는 버튼 생성
        if(i % 7 == 0)
            days[i].setForeground(Color.RED); // 일요일 버튼의 색
        else if(i % 7 == 6)
            days[i].setForeground(Color.BLUE); // 토요일 버튼의 색
        else
            days[i].setForeground(Color.BLACK);
        days[i].addActionListener(this);
        panel.add(days[i]);
    }
    selectPanel.add(chyear);
    selectPanel.add(yLabel);
    selectPanel.add(chmonth);
    selectPanel.add(mLabel);
```

5. Source Code

```
// JPanel bib=new JPanel();
area=new JTextArea(60, 40);
area.setCaretPosition(area.getDocument().getLength());
JScrollPane scrollPane = new JScrollPane(area);
this.add(selectPanel,"North");//연도와 월을 선택할 수 있는 화면을 상단에 출력
this.add(panel, "Center");
this.add(scrollPane,"East");

String m = (ca.get(Calendar.MONTH)+1)+"";
String y = ca.get(Calendar.YEAR)+"";
chyear.select(y);
chmonth.select(m);
chyear.addItemListener(this);
chmonth.addItemListener(this);
}
public void calendar(){
    year = Integer.parseInt(chyear.getSelectedItem());
    month=Integer.parseInt(chmonth.getSelectedItem());
    gc = new GregorianCalendar(year, month-1, 1);
    int max = gc.getActualMaximum(gc.DAY_OF_MONTH);//해당 달의 최대 일 수 획득
    int week = gc.get(gc.DAY_OF_WEEK);//해당 달의 시작 요일
    //System.out.println("DAY_OF_WEEK:"+week);
    String today = Integer.toString(ca.get(Calendar.DATE)); //오늘 날짜 획득
    String today_month = Integer.toString(ca.get(Calendar.MONTH)+1); //오늘의 달 획득
    //System.out.println("today:"+today);
```

6. Source Code

```

for(int i = 0; i < days.length; i++){
    days[i].setEnabled(true);
}
for(int i = 0; i < week-1; i++){//시작일 이전의 버튼을 비활성화
    days[i].setEnabled(false);
}
for(int i = week; i< max+week; i++){
    days[i-1].setText((String.valueOf(i-week+1)));
    days[i-1].setBackground(Color.WHITE);
    if(today_month.equals(String.valueOf(month))){//오늘이 속한 달과 같은 달인 경우
        if(today.equals(days[i-1].getText())){//버튼의 날짜와 오늘날짜가 일치하는 경우
            days[i-1].setBackground(Color.CYAN); //버튼의 배경색 지정
        }
    }
}
for(int i = (max+week-1); i < days.length; i++){//날짜가 없는 버튼을 비활성화
    days[i].setEnabled(false);
}
//System.out.println("max+week:"+ (max+week)+", week:" + week);
}

```

```

@Override
public void itemStateChanged(ItemEvent arg0) {
    Color color = this.getBackground();
    if(arg0.getStateChange()==ItemEvent.SELECTED){
        //년이나 월이 선택되면 기존의 달력을 지우고 새로 그린다.
        for(int i = 0; i < 42; i++){
            if( !days[i].getText().equals("") ){
                days[i].setText(""); //기존의 날짜를 지움
                //달력의 배경색과 동일한 색으로 버튼의 배경색을 설정함.
                days[i].setBackground(color);
            }
        }
        calendar();
    }
}
public static void main(String[] args) {
    new Project02_F();
}
}

```

PROJECT 3 Java Naver Search OpenAPI 활용 Excel 프로젝트

1. Excel 파일 Reading 하기

bookList.xls

Excel API

- > poi-4.1.0.jar - C:\JavaTPC\lib
- > commons-codec-1.12.jar - C:\JavaTPC\lib
- > commons-collections4-4.2.jar - C:\JavaTPC\lib

	A	B	C	D	E	F	G	H	I	J
1	title	ExcelVO	author	company	isbn	imageUrl				
2	물리법칙의 특성		리처드 파인만	해나무	9788956054865	https://bookthumb-phinf.pstatic.net/cover/100/365/10036542.jpg				
3	파이썬을 이용한 빅데이터 수집, 분석과 시각화		이원하	비팬북스	9788994797694	https://bookthumb-phinf.pstatic.net/cover/121/297/12129721.jpg				
4	Java의 정석		남궁성	도우출판	9788994492032	https://bookthumb-phinf.pstatic.net/cover/101/911/10191151.jpg				
5	모듈라 자바		크레이그 월즈	에이콘출판	9788960772359	https://bookthumb-phinf.pstatic.net/cover/067/349/06734990.jpg				

```

import org.apache.poi.hssf.usermodel.*;
import kr.inflearn.ExcelVO;
import java.io.*;
import java.util.*;
public class Project03_A {
    public static void main(String[] args) {
        String filename = "bookList.xls";
        List<ExcelVO> data = new ArrayList<ExcelVO>();
        try (FileInputStream fis = new FileInputStream(filename)) {
            HSSFWorkbook workbook = new HSSFWorkbook(fis);
            HSSFSheet sheet = workbook.getSheetAt(0);
            Iterator rows = sheet.rowIterator();
            String[] imsi=new String[5];
            rows.next();
            while (rows.hasNext()) {
                HSSFRow row = (HSSFRow) rows.next();
                Iterator cells = row.cellIterator();
                int i=0;
                while (cells.hasNext()) {
                    HSSFCell cell = (HSSFCell) cells.next();
                    imsi[i]=cell.toString();
                    i++;
                }
            }
        }
    }
}
  
```



```

public class ExcelVO {
    private String title;
    private String author;
    private String company;
    private String isbn;
    private String imgurl;
    public ExcelVO() { }
}

ExcelVO vo=new ExcelVO(imsi[0],imsi[1],imsi[2],imsi[3], imsi[4]);
data.add(vo);
}//while
} catch (IOException e) {
    e.printStackTrace();
}
showExcelData(data);
}

private static void showExcelData(List<ExcelVO> data) {
    for (ExcelVO vo : data) {
        System.out.println(vo);
    }
}
  
```

1. Excel에 image 저장하기

```

import java.io.*;
import org.apache.poi.hssf.usermodel.HSSFWorkbook;
import org.apache.poi.ss.usermodel.*;
import org.apache.poi.util.IOUtils;
public class Project03_B {
public static void main(String[] args) {
try {
Workbook wb = new HSSFWorkbook();
Sheet sheet = wb.createSheet("My Sample Excel");
InputStream inputStream = new FileInputStream("pic.jpg");
byte[] bytes = IOUtils.toByteArray(inputStream);
int pictureIdx = wb.addPicture(bytes, Workbook.PICTURE_TYPE_JPG);
inputStream.close();

CreationHelper helper = wb.getCreationHelper();
Drawing drawing = sheet.createDrawingPatriarch();

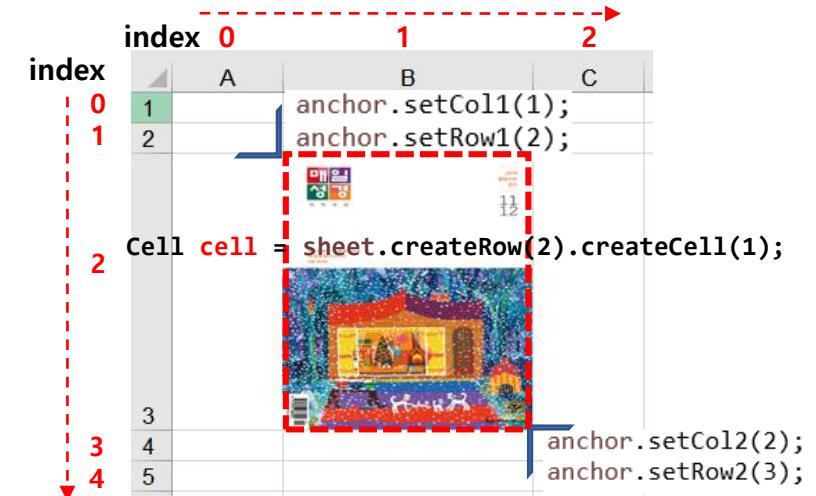
ClientAnchor anchor = helper.createClientAnchor();


anchor.setCol1(1); //Column B
anchor.setRow1(2); //Row 3
anchor.setCol2(2); //Column C
anchor.setRow2(3); //Row 4

}
}
}

```

pic.jpg



```

Picture pict = drawing.createPicture(anchor, pictureIdx);

Cell cell = sheet.createRow(2).createCell(1);
//set width to n character widths = count characters * 256
int widthUnits = 20*256;
sheet.setColumnWidth(1, widthUnits);
//set height to n points in twips = n * 20
short heightUnits = 120*20;
cell.getRow().setHeight(heightUnits);

FileOutputStream fileOut = null;
fileOut = new FileOutputStream("myFile.xlsx");
wb.write(fileOut);
fileOut.close();
} catch (IOException ioex) {
}
}
}

```

1. Excel에 cell DataType 알아보기

```
HSSFCell [cell] = (HSSFCell) cells.next();
CellType type = cell.getCellType();
if (type == CellType.STRING) {
    System.out.println("[ " + cell.getRowIndex() + ", "
        + cell.getColumnIndex() + " ] = STRING; Value = "
        + cell.getRichStringCellValue().toString());
} else if (type == CellType.NUMERIC) {
    System.out.println("[ " + cell.getRowIndex() + ", "
        + cell.getColumnIndex() + " ] = NUMERIC; Value = "
        + cell.getNumericCellValue());
} else if (type == CellType.BOOLEAN) {
    System.out.println("[ " + cell.getRowIndex() + ", "
        + cell.getColumnIndex() + " ] = BOOLEAN; Value = "
        + cell.getBooleanCellValue());
} else if (type == CellType.BLANK) {
    System.out.println("[ " + cell.getRowIndex() + ", "
        + cell.getColumnIndex() + " ] = BLANK CELL");
}
```

cellDataType.xls

	A	B	C	D	E
1	X	2	A	FALSE	B
2	TRUE	1	Y	Z	10

	cell.getRowIndex()	cell.getColumnIndex()
[0, 0]	= STRING; Value = X	
[0, 1]	= NUMERIC; Value = 2.0	
[0, 2]	= STRING; Value = A	
[0, 3]	= BOOLEAN; Value = false	
[0, 4]	= STRING; Value = B	
[1, 0]	= BOOLEAN; Value = true	
[1, 1]	= NUMERIC; Value = 1.0	
[1, 2]	= STRING; Value = Y	
[1, 3]	= STRING; Value = Z	
[1, 4]	= NUMERIC; Value = 10.0	

1. Naver Book OpenAPI 사용 등록하기 <https://developers.naver.com/docs/search/book/>

The screenshot shows the Naver Developers API documentation for the Book Search API. On the left, there's a sidebar with links like 'API 공통 가이드', 'SDK & Tools', 'Clova', '네이버 아이디로 로그인', '파파고', and '서비스 API'. Below that is a login form with fields for '아이디' and '비밀번호' and a '로그인' button. In the center, there's a title '검색 > 책' and a note about REST API and Client ID/Secret. On the right, there's a table with three rows of API details. The first row is for XML output, the second for XML ad output, and the third for JSON output. The 'Request URL' column for the XML output is highlighted with a red dashed box. A red arrow points from the 'Open API Utilization Application' button in the sidebar to the 'Request URL' field in the table.

메서드	인증	요청 URL	출력 포맷	설명
GET	-	https://openapi.naver.com/v1/search/book.xml	XML	책 기본 검색
GET	-	https://openapi.naver.com/v1/search/book_ad.xml	XML	책 상세 검색
GET	-	https://openapi.naver.com/v1/search/book.json	JSON	책 기본 검색

상세 검색은 책 제목(d_title), 저자명(d_auth), 목차(d_cont), ISBN(d_isbn), 출판사(d_publ) 5개 항목 중에서 1개 이상 값을 입력해야 함.

2. Naver Book OpenAPI 사용 등록하기

애플리케이션 등록 (API 이용신청)

애플리케이션의 기본 정보를 등록하면, 좌측 [내 애플리케이션](#) 메뉴의 서브 메뉴에 등록하신 애플리케이션 이름으로 서브 메뉴가 만들어집니다.

애플리케이션 이름 ✓

- 네이버 아이디로 로그인할 때 사용자에게 표시되는 이름이므로 가급적 10자 이내의 간결한 이름을 사용해주세요.
- 40자 이내의 영문, 한글, 숫자, 공백문자, "-", "_" 만 입력 가능합니다.

사용 API ✓

✓

✓

+
✓

비로그인 오픈 API 서비스 환경

✓

등록하기
취소

Application 목록

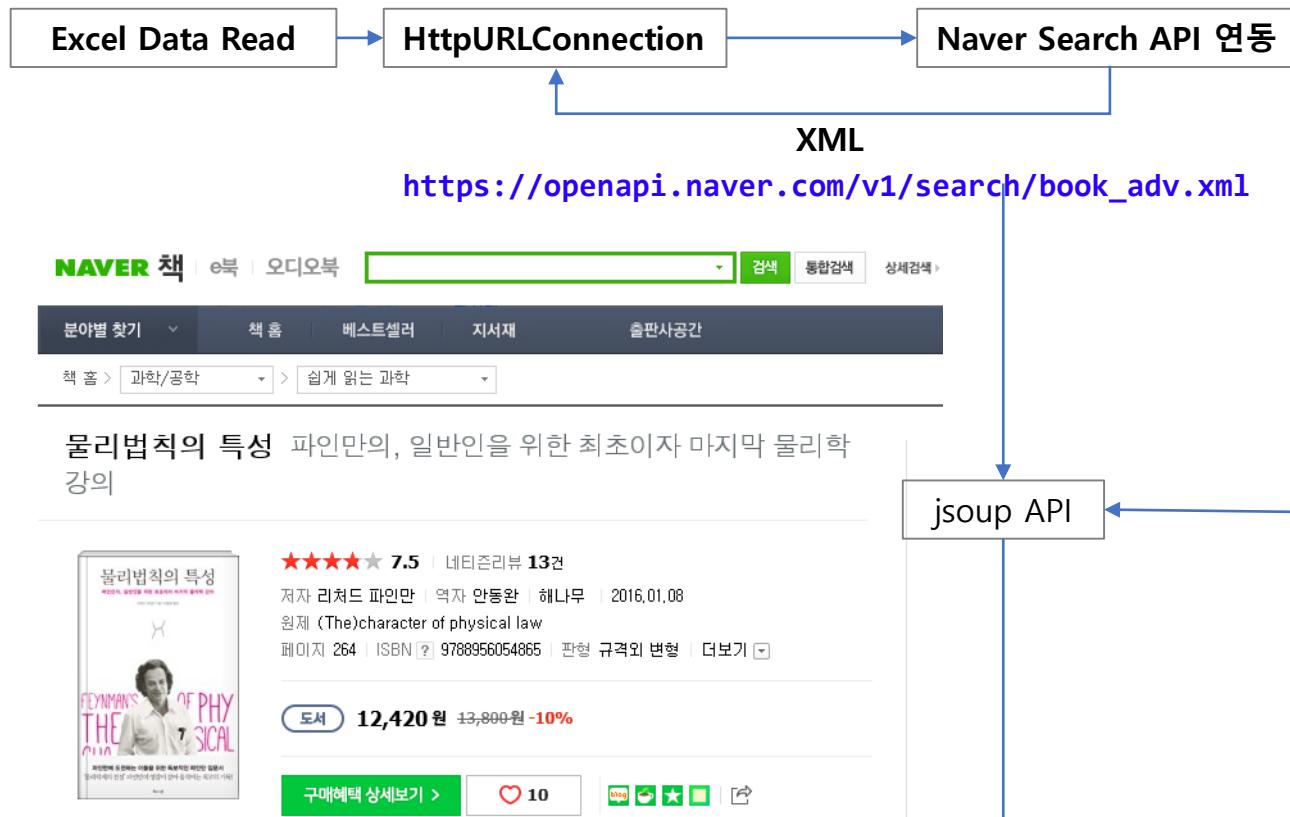
Client ID	Application 명	Action
mJJNeEvkIsiUdn3JDojA	IOT	/
loq4J690dy0Mn36TqjF8	mapService	/
mbVLjPTSAxFPXZOT0IWM	iotService	/
lu8ml6kQy9iv_0g02A3MU	mapview	/
0YpZJ9Y0dxk6bS9rCdt1	myapp	/
p0ZrzVf2ULPCGWlhaKMk	python	/
hQLwYhKiYEGobEsmc8fU	javatpc	/

javatpc

개요	API 설정	멤버관리	로그인 통계	API 통계	Playground(Beta)
애플리케이션 정보					
Client ID	hQLwYhKiYEGobEsmc8fU				
Client Secret	보기			

3. 도서정보를 입력하여 ISBN, Image 검출하기

프로그램 흐름도



실행결과

isbn: 895605486X 9788956054865

<https://bookthumb-phinf.pstatic.net/cover/100/365/10036542.jpg>

```

<!--?xml version="1.0" encoding="UTF-8"?-->
<html>
<head></head>
<body>
<rss version="2.0">
<channel>
<title>Naver Open API - book_adv ::</title>
<link>https://search.naver.com
<description>
    Naver Search Result
</description>
<lastbuilddate>Mon, 05 Aug 2019 10:54:21 +0900</lastbuilddate>
<total>3</total>
<start>1</start>
<display>3</display>
<item>
    <link>http://book.naver.com/bookdb/book_detail.php?bid=10036542
    <img>https://bookthumb-phinf.pstatic.net/cover/100/365/10036542.jpg
    <type=m&update=20160218
    <author>&lt;b&gt;리처드 파인만&lt;/b&gt;</author>
    <price>13800</price>
    <discount>12420</discount>
    <publisher>&lt;b&gt;해나무&lt;/b&gt;</publisher>
    <pubdate>20160108</pubdate>
    <isbn>895605486X 9788956054865</isbn>
</item>
</channel>
</rss>
</body>
</html>
    
```

4. 도서정보를 입력하여 ISBN, Image 검출하기

책제목: Java의 정석
 책저자: 남궁성
 출판사: 도우출판
 isbn: 8994492038 9788994492032
<https://bookthumb-phinf.pstatic.net/cover/101/911/10191151.jpg>
 ExcelVO [title=Java의 정석, author=남궁성, company=도우출판, isbn=9788994492032, imgurl=10191151.jpg]

```
Document doc = Jsoup.parse(response.toString());
//System.out.println(doc.toString());
Element total = doc.select("total").first();
if(!(total.text().equals("0"))) {
    Element isbn = doc.select("isbn").first();
    System.out.println("isbn: "+isbn.text());
    String img=doc.toString();
    String imgTag=img.substring(img.indexOf("<img>")+5);
    img=imgTag.substring(0, imgTag.indexOf("?"));
    System.out.println(img);
    vo.setIsbn(isbn.text().split(" ")[1]);
    String fileName=img.substring(img.lastIndexOf("/") + 1);
    vo.setImgurl(fileName);
    System.out.println(vo);
}
else
    System.out.println("검색데이터가 없습니다.");
```

```
String URL_STATICMAP = "https://openapi.naver.com/v1/search/book_adv.xml?d_titl=" +
URLEncoder.encode(vo.getTitle(), "UTF-8") +
"&d_auth=" + URLEncoder.encode(vo.getAuthor(), "UTF-8") +
"&d_publ=" + URLEncoder.encode(vo.getCompany(), "UTF-8");
URL url = new URL(URL_STATICMAP);
HttpURLConnection con = (HttpURLConnection)url.openConnection();
con.setRequestMethod("GET");
con.setRequestProperty("X-Naver-Client-Id", "p0ZrzVf2ULPCGWhdaKMk");
con.setRequestProperty("X-Naver-Client-Secret", "R7H5MMEDEO");
```

```
<item>
<title>&lt;b&gt;Java의 정석&lt;/b&gt; (최신 &lt;b&gt;Java&lt;/b&gt; 8.0 포함)</title>
<link>https://book.naver.com/bookdb/book\_detail.php?bid=10191151</link>
<img>https://bookthumb-phinf.pstatic.net/cover/101/911/10191151.jpg?type=m1&update=20190204</img>
<author>
&lt;b&gt;남궁성&lt;/b&gt;
</author>
<price>
30000
</price>
<discount>
27000
</discount>
<publisher>
&lt;b&gt;도우출판&lt;/b&gt;
</publisher>
<pubdate>
20160127
</pubdate>
<isbn>
8994492038 9788994492032
</isbn>
<description>
자바의 기초부터 실전활용까지 모두 담다! 자바의 기초부터 객제지향개념을 넘어 실전활용까지 수록한 <b>JAVA의 정석</b> . 저자의 .
</description>
</item>
```

1. 도서정보를 활용 ISBN, Image 검출하여 Excel에 저장하기

```
입력처리(I)/종료(E):I
책제목: 물리법칙의 특성
책저자: 리처드 파인만
출판사: 해나무
isbn: 895605486X 9788956054865
https://bookthumb-phinf.pstatic.net/cover/100/365/10036542.jpg
계속입력 하시면 Y / 입력종료 N:download complete...
10036542.jpg
Y
책제목: 파이썬을 이용한 빅데이터 수집, 분석과 시각화
책저자: 이원하
출판사: 비팬북스
isbn: 8994797696 9788994797694
https://bookthumb-phinf.pstatic.net/cover/121/297/12129721.jpg
계속입력 하시면 Y / 입력종료 N:download complete...
12129721.jpg
Y
책제목: Java의 정석
책저자: 남궁성
출판사: 도우출판
isbn: 8994492038 9788994492032
https://bookthumb-phinf.pstatic.net/cover/101/911/10191151.jpg
계속입력 하시면 Y / 입력종료 N:download complete...
10191151.jpg
Y
책제목: 모듈라 자바
책저자: 크레이그 월즈
출판사: 에이콘출판
isbn: 8960772356 9788960772359
https://bookthumb-phinf.pstatic.net/cover/067/349/06734990.jpg
계속입력 하시면 Y / 입력종료 N:download complete...
06734990.jpg
N
데이터 추출중.....
ISBN, ImageURL 저장성공
```



isbn.xls



A	B	C	D	E	F
1 책제목	저자	출판사	isbn	이미지이름	이미지
2 물리법칙의 특성	리처드 파인만	해나무	9788956054865	10036542.jpg	
3 파이썬을 이용한 빅데이터 수집, 분석과 시각화	이원하	비팬북스	9788994797694	12129721.jpg	
4 Java의 정석	남궁성	도우출판	9788994492032	10191151.jpg	

NAVER 책 | e북 | 오디오북 | 검색 | 통합검색 | 상세검색

분야별 찾기 | 책 홈 | 베스트셀러 | 지서재 | 출판사공간

책 홈 > 과학/공학 > 쉽게 읽는 과학

물리법칙의 특성 파인만의, 일반인을 위한 최초이자 마지막 물리학 강의

★★★★★ 7.5 | 네이버리뷰 13건

저자 리처드 파인만 | 역사 안동완 | 해나무 | 2016.01.08

원제 (The)character of physical law
페이지 264 ISBN ? 9788956054865 판형 규격외 변형 | 더보기

도서 12,420 원 13,800 원 -10%

구매혜택 상세보기 > 10 |

데이터 추출중....
ISBN, ImageURL 저장성공

1. Source Code

```

import java.io.BufferedReader;
import java.io.InputStreamReader;
import kr.inflearn.ExcelDAO;

public class Project03_F {
    public static void main(String[] args) {
        ExcelDAO dao=new ExcelDAO();
        BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
        try {
            System.out.print("입력처리(I)/종료(E):");
            String sw=br.readLine();
            switch (sw) {
                case "I":
                    dao.excel_input();
                    break;
                case "E":
                    System.out.println("프로그램종료");
                    System.exit(0);
                    break;
                default:
                    System.out.println("I or E input");
            }
        }catch(Exception e) {
            e.printStackTrace();
        }
    }
}

```

입력처리(I)/종료(E):I
 책제목: 물리 법칙의 특성
 책저자: 리처드 파인만
 출판사: 해나무
 isbn: 895605486X 9788956054865
<https://bookthumb-phinf.pstatic.net/cover/100/365/10036542.jpg>
 계속입력 하시면 Y / 입력종료 N:download complete...
 10036542.jpg
 Y
 책제목: 파이썬을 이용한 빅데이터 수집, 분석과 시각화
 책저자: 이원하
 출판사: 비팬북스
 isbn: 8994797696 9788994797694
<https://bookthumb-phinf.pstatic.net/cover/121/297/12129721.jpg>
 계속입력 하시면 Y / 입력종료 N:download complete...
 12129721.jpg
 Y
 책제목: Java의 정석
 책저자: 남궁성
 출판사: 도우출판
 isbn: 8994492038 9788994492032
<https://bookthumb-phinf.pstatic.net/cover/101/911/10191151.jpg>
 계속입력 하시면 Y / 입력종료 N:download complete...
 10191151.jpg
 Y
 책제목: 모듈라 자바
 책저자: 크레이그 컬즈
 출판사: 에이콘출판
 isbn: 8960772356 9788960772359
<https://bookthumb-phinf.pstatic.net/cover/067/349/06734990.jpg>
 계속입력 하시면 Y / 입력종료 N:download complete...
 06734990.jpg
 N
 데이터 추출중.....
 ISBN, ImageURL 저장성공

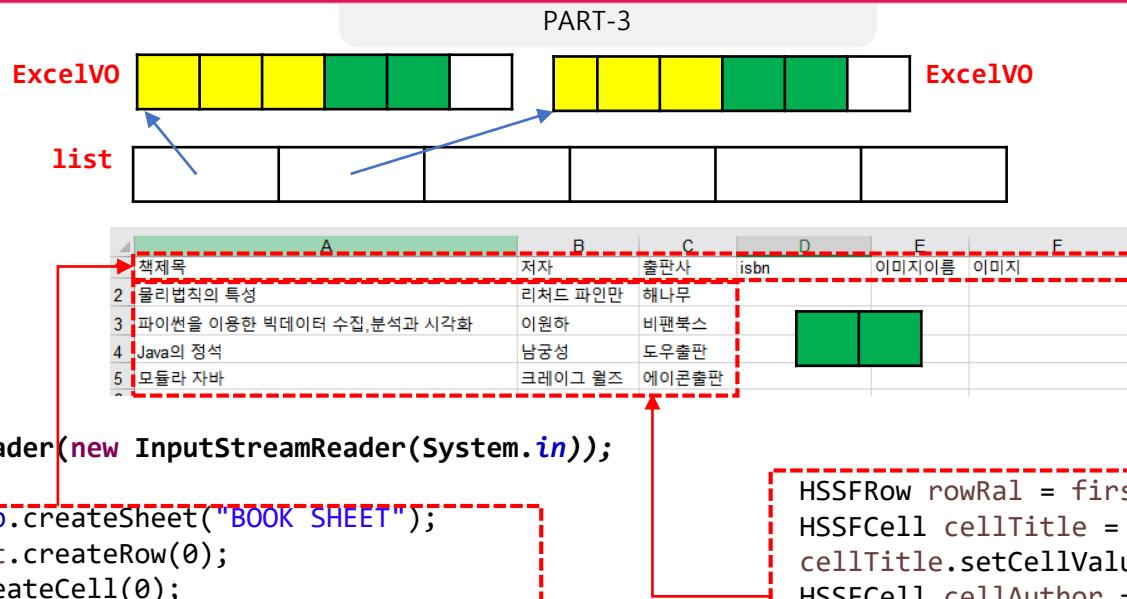
10. Java Excel Handling(STEP 5)

2. Source Code

```

package kr.inflearn;
public class ExcelDAO {
private List<ExcelVO> list;
private HSSFWorkbook wb;
public ExcelDAO() {
    list=new ArrayList<ExcelVO>();
    wb=new HSSFWorkbook();
}
public void excel_input() {
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
try {
    HSSFSheet firstSheet = wb.createSheet("BOOK SHEET");
    HSSFRow rowA = firstSheet.createRow(0);
    HSSFCell cellA = rowA.createCell(0);
    cellA.setCellValue(new HSSFRichTextString("책제목"));
    HSSFCell cellB = rowA.createCell(1);
    cellB.setCellValue(new HSSFRichTextString("저자"));
    HSSFCell cellC = rowA.createCell(2);
    cellC.setCellValue(new HSSFRichTextString("출판사"));
    HSSFCell cellD = rowA.createCell(3);
    cellD.setCellValue(new HSSFRichTextString("isbn"));
    HSSFCell cellE = rowA.createCell(4);
    cellE.setCellValue(new HSSFRichTextString("이미지이름"));
    HSSFCell cellF = rowA.createCell(5);
    cellF.setCellValue(new HSSFRichTextString("이미지"));
    int i=1;
    while(true) {
        System.out.print("책제목:");
        String title=br.readLine();
        System.out.print("책저자:");
        String author=br.readLine();
        System.out.print("출판사:");
        String company=br.readLine();
        ExcelVO vo=new ExcelVO(title, author, company);
        // isbn, image 검색
        ExcelVO data=naver_search(vo);
        list.add(data);
        System.out.print("계속입력 하시면 Y / 입력종료 N:");
        String key=br.readLine();
        if(key.equals("N")) break;
    }
    System.out.println("데이터 추출중.....");
    excel_save();
} catch (Exception e) {
    e.printStackTrace();
}
}
}

```



```

public class ExcelVO {
    private String title;
    private String author;
    private String company;
    private String isbn;
    private String imgurl;
    public ExcelVO() { }
}

HSSFRow rowR1 = firstSheet.createRow(i);
HSSFCell cellTitle = rowR1.createCell(0);
cellTitle.setCellValue(new HSSFRichTextString(title));
HSSFCell cellAuthor = rowR1.createCell(1);
cellAuthor.setCellValue(new HSSFRichTextString(author));
HSSFCell cellCompany = rowR1.createCell(2);
cellCompany.setCellValue(new HSSFRichTextString(company));
i++;
ExcelVO vo=new ExcelVO(title, author, company);
// isbn, image 검색
ExcelVO data=naver_search(vo);
list.add(data);
System.out.print("계속입력 하시면 Y / 입력종료 N:");
String key=br.readLine();
if(key.equals("N")) break;
}
System.out.println("데이터 추출중.....");
excel_save();
} catch (Exception e) {
    e.printStackTrace();
}
}

```

3. Source Code

//상세 검색은 책 제목(d_titl), 저자명(d_auth), 목차(d_cont), ISBN(d_isbn), 출판사(d_publ) 5개 항목 중에서 1개 이상 값을 입력해야함.

```
public ExcelVO naver_search(ExcelVO vo) {
try {
String URL_STATICMAP = "https://openapi.naver.com/v1/search/book_adv.xml?d_titl="+URLEncoder.encode(vo.getTitle(), "UTF-
8")+"&d_auth="+URLEncoder.encode(vo.getAuthor(), "UTF-8")+"&d_publ="+URLEncoder.encode(vo.getCompany(), "UTF-8");
URL url = new URL(URL_STATICMAP);
HttpURLConnection con = (HttpURLConnection)url.openConnection();
con.setRequestMethod("GET");
con.setRequestProperty("X-Naver-Client-Id", "p0ZrzVf2ULPCGWhdaKMk");
con.setRequestProperty("X-Naver-Client-Secret", "R7H5MMEDEO");
int responseCode = con.getResponseCode();
BufferedReader br;
if(responseCode==200) {
    br = new BufferedReader(new InputStreamReader(con.getInputStream(),"UTF-8"));
} else {
    br = new BufferedReader(new InputStreamReader(con.getErrorStream()));
}
String inputLine;
StringBuffer response = new StringBuffer(); //문자열 추가 변경시 사용
while ((inputLine = br.readLine()) != null) {
    response.append(inputLine);
}
br.close();
}
```

ExcelVO



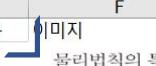
```
Document doc = Jsoup.parse(response.toString());
//System.out.println(doc.toString());
Element isbn = doc.select("isbn").first();
System.out.println("isbn: "+ isbn.text());
String img=doc.toString();
String imgTag=img.substring(img.indexOf("<img>")+5);
img=imgTag.substring(0, imgTag.indexOf("?"));
System.out.println(img);
vo.setIsbn(isbn.text().split(" ")[1]);
String fileName=img.substring(img.lastIndexOf("/") + 1);
vo.setImgurl(fileName);
// DownloadBroker
Runnable dl=new DownloadBroker(img, fileName);
Thread t=new Thread(dl);
t.start();
} catch (Exception e) {
    System.out.println(e);
}
return vo;
}//naver_search
```

4. Source Code

```
public void excel_save() {  
    try {  
        HSSFSheet sheet = wb.getSheetAt(0);  
        if(wb != null && sheet != null) {  
            Iterator rows = sheet.rowIterator();  
            rows.next();  
            int i=0; // list의 index  
            while (rows.hasNext()) {  
  
                HSSFRow row = (HSSFRow) rows.next();  
                HSSFCell cell=row.createCell(3);  
  
                cell.setCellType(CellType.STRING);  
                cell.setCellValue(list.get(i).getIsbn());  
                cell=row.createCell(4);  
                cell.setCellType(CellType.STRING);  
                cell.setCellValue(list.get(i).getImgurl());  
  
                InputStream inputStream = new FileInputStream(list.get(i).getImgurl());  
                byte[] bytes = IOUtils.toByteArray(inputStream);  
                int pictureIdx = wb.addPicture(bytes, Workbook.PICTURE_TYPE_JPEG);  
                inputStream.close();  
  
                CreationHelper helper = wb.getCreationHelper();  
                Drawing drawing = sheet.createDrawingPatriarch();  
                ClientAnchor anchor = helper.createClientAnchor();  
            }  
        }  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
}
```

```
anchor.setCol1(5);
anchor.setRow1(i+1);
anchor.setCol2(6);
anchor.setRow2(i+2);
Picture pict = drawing.createPicture(anchor, pictureIdx);
Cell cellImg = row.createCell(5);
int widthUnits = 20*256;
sheet.setColumnWidth(5, widthUnits);
short heightUnits = 120*20; // 1/20
cellImg.getRow().setHeight(heightUnits);
i++;
}
FileOutputStream fos = new FileOutputStream("isbn.xls");
wb.write(fos);
fos.close();
System.out.println("ISBN, ImageURL 저장성공");
}
}catch(Exception e) {
e.printStackTrace();
}
```

PART-3

	A	B	C	D	E	F	G
0	책제목	저자	출판사	isbn	이미지이름	이미지	물리법칙의 특성
1							

PROJECT 4 Java iText API 활용 PDF 프로젝트

PART-3

1. iText API를 이용한 PDF table 만들기

TABLE 만들기

```
PdfPTable table = new PdfPTable()
table.setWidthPercentage(100); // table의 폭%
```

Table Column 폭 조절

```
float[] columnWidths = new float[]{20f, 15f, 15f, 30f};
table.setWidths(columnWidths);
```

Table Cell 정렬

H정렬(Element.ALIGN_LEFT, Element.ALIGN_RIGHT, Element.ALIGN_CENTER)
`cell.setHorizontalAlignment(Element.ALIGN_CENTER);`

V정렬(Element.ALIGN_TOP, Element.ALIGN_MIDDLE, Element.ALIGN_BOTTOM)
`cell.setVerticalAlignment(Element.ALIGN_MIDDLE);`

Table Cell 여백조절

```
cell.setPadding(10);
cell.setPaddingTop(20);
cell.setPaddingRight(30);
cell.setPaddingBottom(20);
cell.setPaddingLeft(30);
```

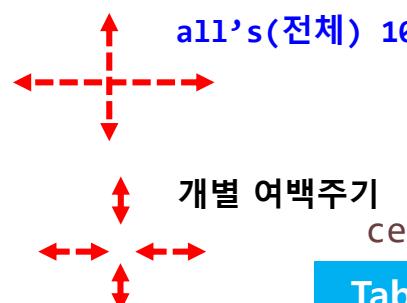


Table Column 합치기

PdfPCell

구절 Phrase

제목	저자	출판사	이미지URL
물리법칙의 특성 파이썬을 이용한 빅데이터 수집, 분석과 시각화	리처드 파인만 이원하	해나무 비팬북스	https://bookthumb-phinf.pstatic.net/cover/100/365/10036542.jpg https://bookthumb-phinf.pstatic.net/cover/121/297/12129721.jpg
Java의 정석	남궁성	도우출판	https://bookthumb-phinf.pstatic.net/cover/101/911/10191151.jpg

제목	저자	출판사	이미지URL
물리법칙의 특성	리처드 파인만	해나무	https://bookthumb-phinf.pstatic.net/cover/100/365/10036542.jpg
파이썬을 이용한 빅데이터 수집, 분석과 시각화	이원하	비팬북스	https://bookthumb-phinf.pstatic.net/cover/121/297/12129721.jpg
Java의 정석	남궁성	도우출판	https://bookthumb-phinf.pstatic.net/cover/101/911/10191151.jpg

제목	저자	출판사	이미지URL
물리법칙의 특성	리처드 파인만	해나무	https://bookthumb-phinf.pstatic.net/cover/100/365/10036542.jpg
파이썬을 이용한 빅데이터 수집, 분석과 시각화	이원하	비팬북스	https://bookthumb-phinf.pstatic.net/cover/121/297/12129721.jpg
Java의 정석	남궁성	도우출판	https://bookthumb-phinf.pstatic.net/cover/101/911/10191151.jpg

1. iText API를 이용한 PDF에 Paragraph(단락, 문단, 절) 만들기

Paragraph(단락, 문단, 절) 만들기

```
Paragraph paragraph = new Paragraph();
```

Chunk(부분, 규모가 큰) 만들기

```
String content="Your word is a lamp to my feet and a light for my path";  
Chunk chunk = new Chunk(content);
```

Paragraph 줄 간격

```
String content="Your word is a lamp to my feet and a light for my path";
Paragraph para1 = new Paragraph(32);
para1.setSpacingBefore(50);
para1.setSpacingAfter(50);

for (int i = 0; i < 20; i++) {
    Chunk chunk = new Chunk(content);
    para1.add(chunk);
}
doc.add(para1);

Paragraph para2 = new Paragraph();
for (int i = 0; i < 10; i++) {
    para2.add(new Chunk(content));
}
doc.add(para2);
```

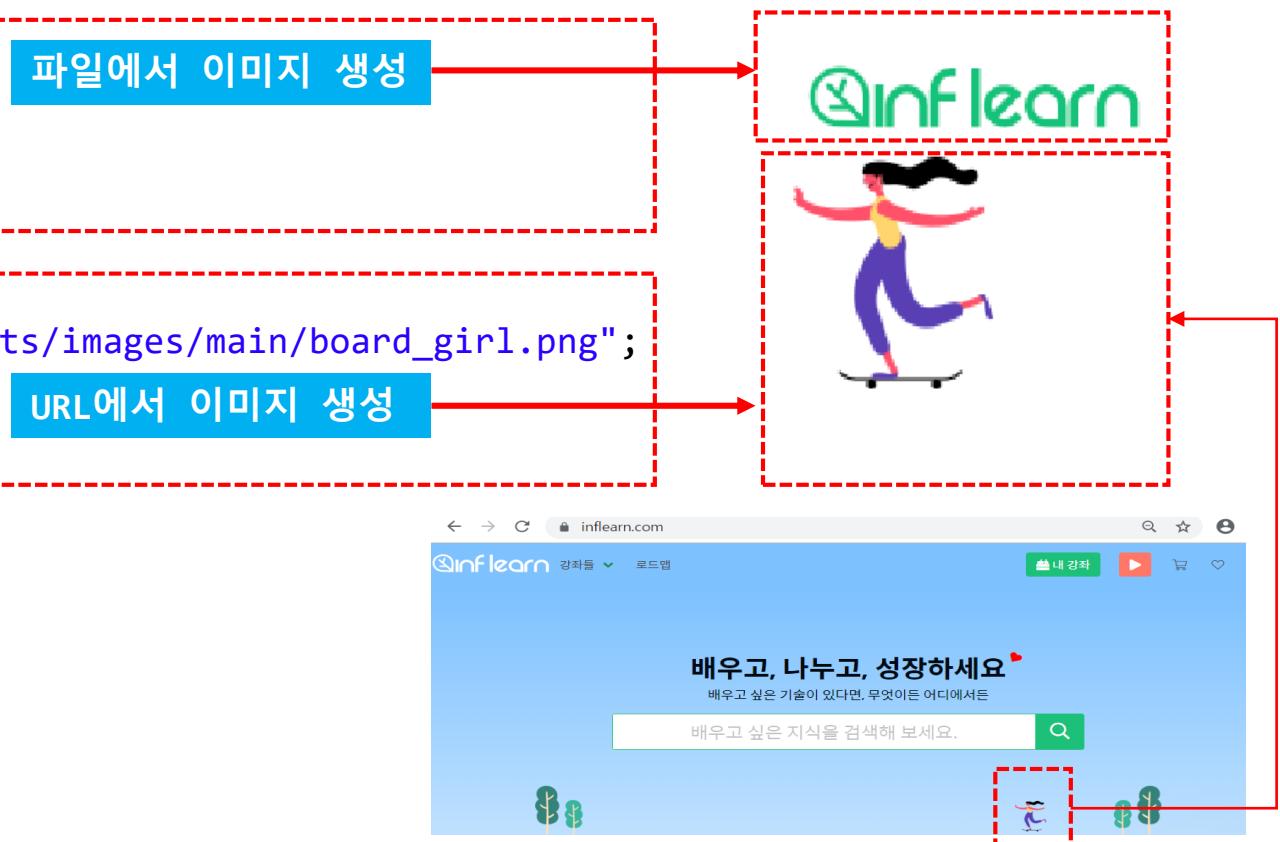
1. iText API를 이용한 PDF에 image 삽입하기

```
Document doc = new Document();
try {
    PdfWriter.getInstance(doc, new FileOutputStream("ImageDemo.pdf"));
    doc.open();

    // 파일에서 이미지 생성
    String filename = "inflearn.png";
    Image image = Image.getInstance(filename);
    doc.add(image);

    // URL로 부터 이미지 생성
    String url = "https://cdn.inflearn.com/assets/images/main/board_gi
    image = Image.getInstance(url);
    doc.add(image);

} catch (Exception e) {
    e.printStackTrace();
} finally {
    doc.close();
}
```



1. iText API를 이용한 PDF에 image 크기조절

```

PdfWriter.getInstance(doc, new FileOutputStream("ImageScaling.pdf"));
doc.open();

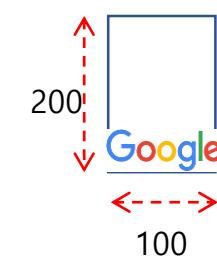
String filename = "googlelogo.png";
Image image = Image.getInstance(filename);
doc.add(image);

String filename1 = "googlelogo.png";
Image image1 = Image.getInstance(filename1);
image1.scaleAbsolute(200f, 200f);
doc.add(image1);

String url =
"https://www.google.com/images/branding/googlelogo/1x/googlelogo_color_272x92dp.png";
image = Image.getInstance(url);
image.scalePercent(200f);
doc.add(image);

image = Image.getInstance(url);
image.scaleToFit(100f, 200f);
doc.add(image);

```



1. Excel에서 데이터를 읽어서 PDF로 만들기

입력처리(I)/종료(E):I

책제목: Java의 정석

책저자: 남궁성

출판사: 도우출판

isbn: 8994492038 9788994492032

<https://bookthumb-phinf.pstatic.net/cover/101/911/10191151.jpg>

계속입력 하시면 Y / 입력종료 N:download complete...

10191151.jpg

Y

책제목: 리눅스 프로그래밍

책저자: 창병모

출판사: 생능출판

isbn: 8970508155 9788970508153

<https://bookthumb-phinf.pstatic.net/cover/078/200/07820067.jpg>

계속입력 하시면 Y / 입력종료 N:download complete...

07820067.jpg

Y

책제목: 쉽게 배우는 R 데이터 분석

책저자: 김영우

출판사: 이지스퍼블리싱

isbn: 1187370940 9791187370949

<https://bookthumb-phinf.pstatic.net/cover/122/565/12256508.jpg>

계속입력 하시면 Y / 입력종료 N:download complete...

12256508.jpg

N

데이터 추출중.....

ISBN, ImageURL 저장성공

A	B	C	D	E	F
책제목	저자	출판사	isbn	이미지이름	이미지
Java의 정석	남궁성	도우출판	9788994492032	10191151.jpg	
리눅스 프로그래밍	창병모	생능출판	9788970508153	07820067.jpg	
쉽게 배우는 R 데이터 분석	김영우	이지스퍼블리싱	9791187370949	12256508.jpg	

isbn.xls

제목	저자	출판사	이미지
Java의 정석	남궁성	도우출판	
리눅스 프로그래밍	창병모	생능출판	
쉽게 배우는 R 데이터 분석	김영우	이지스퍼블리싱	

bookList.pdf

Excel

PDF

2. Source Code

```

public class Project04_F {
public static void main(String[] args) {
String filename = "isbn.xls";
List<ExcelVO> data = new ArrayList<ExcelVO>();
try (FileInputStream fis = new FileInputStream(filename)) {
HSSFWorkbook workbook = new HSSFWorkbook(fis);
HSSFSheet sheet = workbook.getSheetAt(0);
Iterator rows = sheet.rowIterator();
String[] imsi=new String[5];
rows.next();
while (rows.hasNext()) {
HSSFRow row = (HSSFRow) rows.next();
Iterator cells = row.cellIterator();
int i=0;
while (cells.hasNext()) {
HSSFCell cell = (HSSFCell) cells.next();
imsi[i]=cell.toString();
i++;
if(i==5) break;
}
ExcelVO vo=new ExcelVO(imsi[0],imsi[1],imsi[2],imsi[3], imsi[4]);
data.add(vo);
}
pdf_maker(data);
} catch (IOException e) {
e.printStackTrace();
}
}

```

```

import java.io.*;
import java.util.ArrayList;
import java.util.Iterator;
import java.util.List;
import org.apache.poi.hssf.usermodel.*;
import com.itextpdf.text.*;
import com.itextpdf.text.pdf.*;
import kr.inflearn.ExcelVO;

```

isbn.xls

A	B	C	D	E	F
책제목	저자	출판사	isbn	이미지이름	이미지
2 Java의 정석	남궁성	도우출판	9788994492032	10191151.jpg	
3 리눅스 프로그래밍	창 병모	생능출판	9788970508153	07820067.jpg	
4 쉽게 배우는 R 데이터 분석	김영우	이지스퍼블리싱	9791187370949	12256508.jpg	

3. Source Code

```
private static void pdf_maker(List<ExcelVO> data) {  
    String[] headers = new String[]{ "제목", "저자", "출판사", "이미지" };  
    Document document = new Document(PageSize.A4);  
    try {  
        PdfWriter.getInstance(document, new FileOutputStream(new File("bookList.pdf")));  
        document.open();  
  
        BaseFont bFont=BaseFont.createFont("MALGUN.TTF",  
                                         BaseFont.IDENTITY_H, BaseFont.NOT_EMBEDDED);  
        Font fontHeader=new Font(bFont, 12);  
        Font fontRow=new Font(bFont, 10);  
        PdfPTable table = new PdfPTable(headers.length);  
        for (String header : headers) {  
            PdfPCell cell = new PdfPCell();  
            cell.setGrayFill(0.9f);  
            cell.setPhrase(new Phrase(header.toUpperCase(), fontHeader));  
            table.addCell(cell);  
        }  
        table.completeRow();  
    }
```

```
for(ExcelVO vo : data) {  
  
    Phrase phrase = new Phrase(vo.getTitle(), fontRow);  
    table.addCell(new PdfPCell(phrase));  
  
    phrase = new Phrase(vo.getAuthor(), fontRow);  
    table.addCell(new PdfPCell(phrase));  
  
    phrase = new Phrase(vo.getCompany(), fontRow);  
    table.addCell(new PdfPCell(phrase));  
  
    Image image = Image.getInstance(vo.getImgurl());  
    table.addCell(image);  
  
    table.completeRow();  
}  
document.addTitle("PDF Table Demo");  
document.add(table);  
System.out.println("bookList 생성완료");  
}catch(Exception e) {  
    e.printStackTrace();  
}finally {  
    document.close();  
}  
}
```

PROJECT 5 Java MQTT Client 만들기(온.습도 모니터링 및 LED제어)

1. Eclipse Mosquitto™ An open source MQTT broker Download

→ <https://mosquitto.org/>



Eclipse Mosquitto is an open source (EPL/EDL licensed) message broker that implements the MQTT protocol. It is designed to be highly portable and available for a wide range of platforms. Go to the dedicated download page to find the source code or binaries for your platform.

The Mosquitto project also provides a C library for implementing MQTT clients, and the very popular Mosquitto is part of the Eclipse Foundation, is an iot.eclipse.org project and is sponsored by ced

Test

You can have your own instance of Mosquitto running in minutes, but to make testing even easier, the Mosquitto Project runs a test server at test.mosquitto.org where you can test your clients in a variety of environments.

Download

Mosquitto is highly portable and available for a wide range of platforms. Go to the dedicated [download page](#) to find the source code or binaries for your platform.

Test

You can have your own instance of Mosquitto running in minutes, but to make testing even easier, the Mosquitto Project runs a test server at test.mosquitto.org where you can test your clients in a variety of environments.



Source

- [mosquitto-1.6.6.tar.gz \(319kB\) \(GPG signature\)](#)
- [Git source code repository \(github.com\)](#)

Older downloads are available at <https://mosquitto.org/files/>

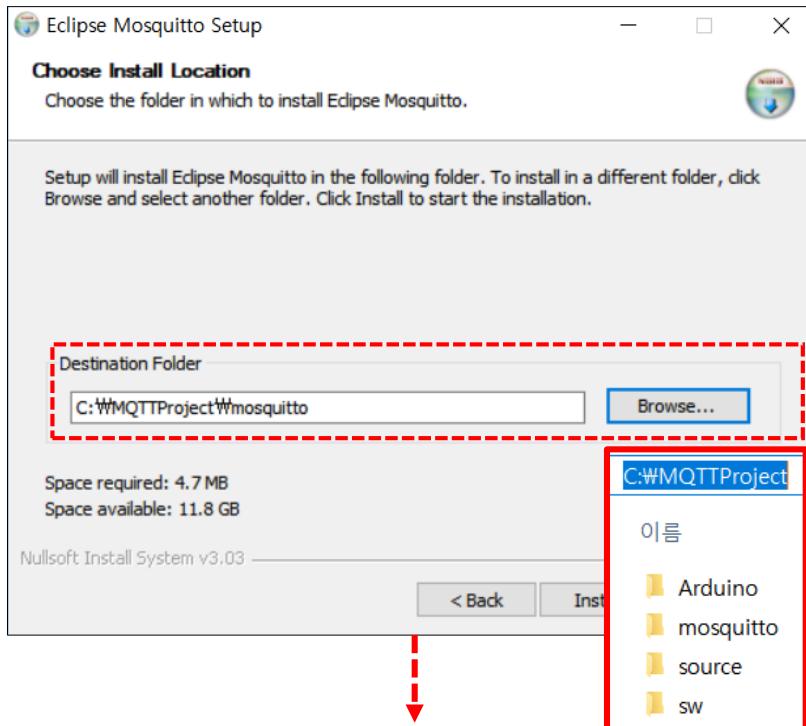
Binary Installation

The binary packages listed below are supported by the Mosquitto project. In ma

Windows

- [mosquitto-1.6.6-install-windows-x64.exe \(~1.4 MB\) \(64-bit build, Windows\)](#)
- [mosquitto-1.6.6-install-windows-x32.exe \(~1.4 MB\) \(32-bit build, Windows\)](#)

2. Mosquitto™ MQTT broker 설치 및 서버구동



① cmd(관리자 권한으로 실행)에서 서버 구동 →서버 구동 창

```
C:\MQTTProject\mosquitto>mosquitto -v
```

```
1569209814: mosquitto version 1.6.6 starting
```

```
1569209814: Using default config.
```

```
1569209814: Opening ipv6 listen socket on port 1883.
```

```
1569209814: Opening ipv4 listen socket on port 1883.
```

mosquito server 구동 및 구독자, 발행자 실행

② subscriber(구독자) 실행→수신대기 창

```
C:\MQTTProject\mosquitto>mosquitto_sub -t iot -p 1883
hello
{"tmp":25,"hum":70}
```

(외부에서 연결하는 방법)

```
C:\MQTTProject\mosquitto>mosquitto_sub -h MQTT서버ip주소 -t iot -p 1883
```

③ publisher(발행자) 실행→메시지(토픽) 발행 창

```
C:\MQTTProject\mosquitto>mosquitto_pub -t iot -m "hello"
```

```
C:\MQTTProject\mosquitto>mosquitto_pub -t iot -m "{\"tmp\":25,\"hum\":70}"
```

3. MQTT broker 메시지 중개 테스트

The image shows three terminal windows illustrating MQTT communication:

- Top Left Terminal:** Shows the MQTT broker (mosquitto) starting up. It receives a connection from an Arduino client (mosq/c14VD7MdHjr05rT1s) and sends a CONNACK message.
- Top Right Terminal:** Shows a client (mosquitto_sub) subscribed to the "iot" topic. It receives two JSON messages from the broker: "hello" and a temperature-humidity pair ("tmp": 25, "hum": 70).
- Bottom Terminal:** Shows a client (mosquitto_pub) publishing a "hello" message to the "iot" topic. It then publishes a JSON message ("tmp": 25, "hum": 70) to the same topic.

Annotations highlight specific parts of the communication:

- A red dashed box surrounds the broker's startup log in the top-left terminal.
- A red box labeled "구독" (Subscribe) highlights the subscription command in the top-right terminal.
- A yellow box labeled "JSON" highlights the JSON data being published in the bottom terminal.
- A red box labeled "발행" (Publish) highlights the publish command in the bottom terminal.

```

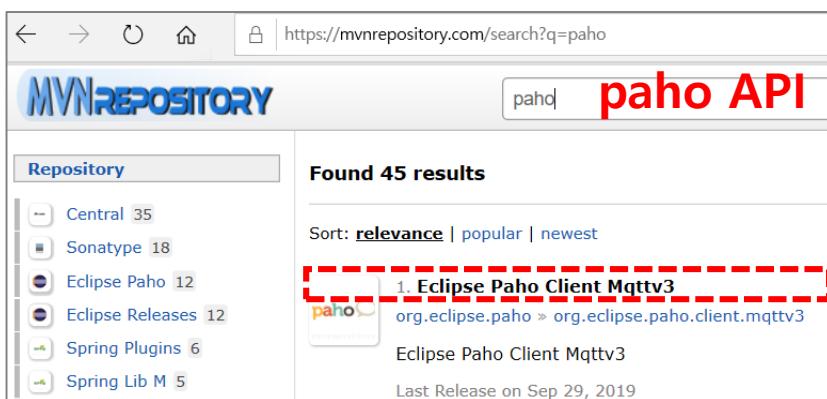
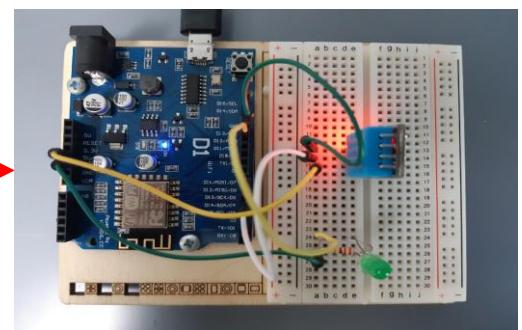
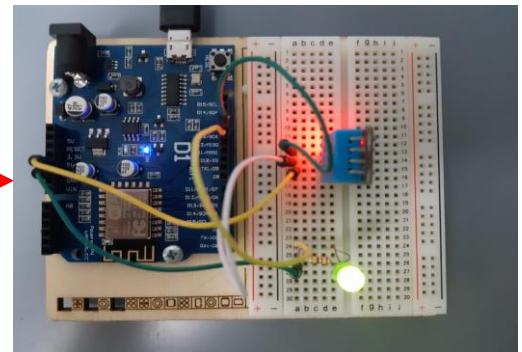
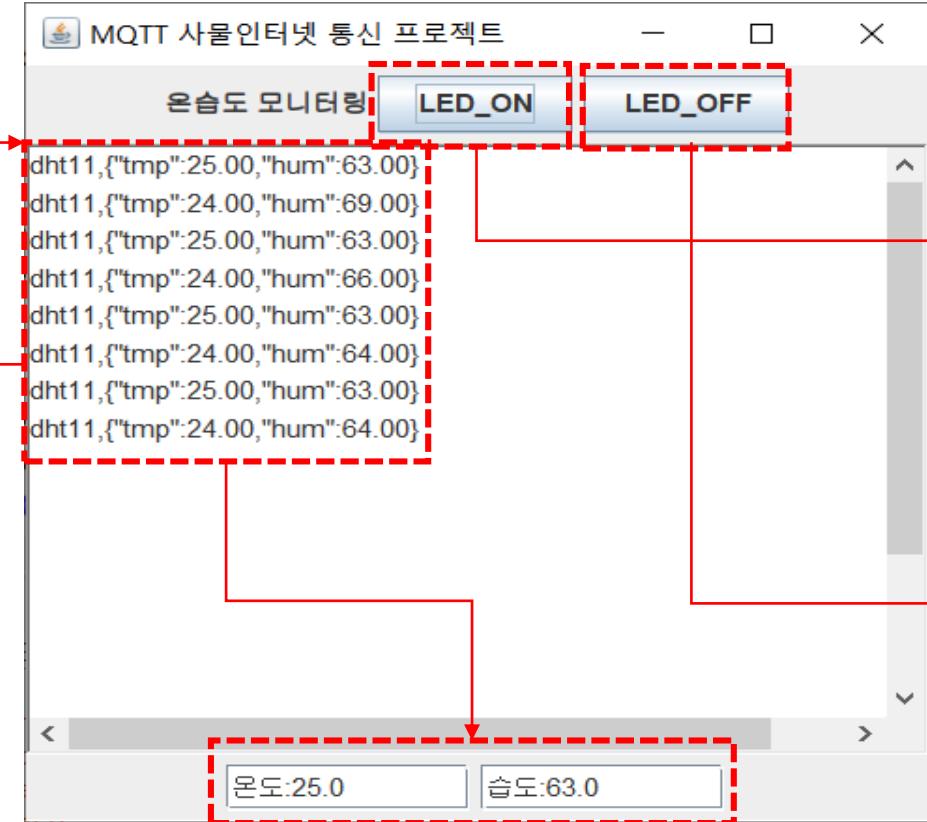
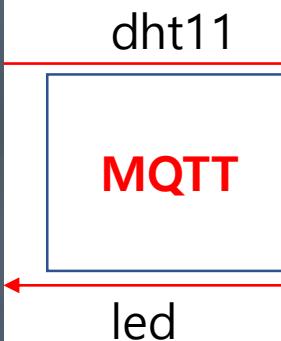
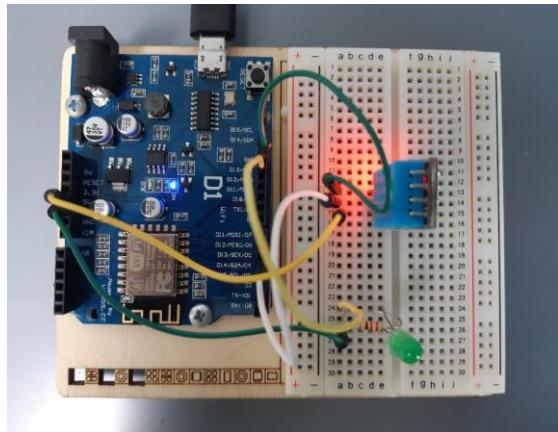
C:\MQTTProject\mosquitto>mosquitto -v
1569209814: mosquitto version 1.6.6 starting
1569209814: Using default config.
1569209814: Opening ipv6 listen socket on port 1883.
1569209814: Opening ipv4 listen socket on port 1883.
1569210272: New connection from ::1 on port 1883.
1569210272: New client connected from ::1 as mosq/c14VD7MdHjr05rT1s (p2, c1, k60).
1569210272: No will message specified.
1569210272: Sending CONNACK to mosq/c14VD7MdHjr05rT1s (0, 0)
1569210272: Received SUBSCRIBE from mosq/c14VD7MdHjr05rT1s
1569210272:     iot (QoS 0)
1569210272: mosq/c14VD7MdHjr05rT1s 0 iot
1569210272: Sending SUBACK to mosq/c14VD7MdHjr05rT1s
1569210332: Received PINGREQ from mosq/c14VD7MdHjr05rT1s
1569210332: Sending PINGRESP to mosq/c14VD7MdHjr05rT1s
1569210392: Received PINGREQ from mosq/c14VD7MdHjr05rT1s
  
```

Arduino(WeMos)쪽에서 WiFi로 DHT11 Sensor 데이터(온도/습도) 발행

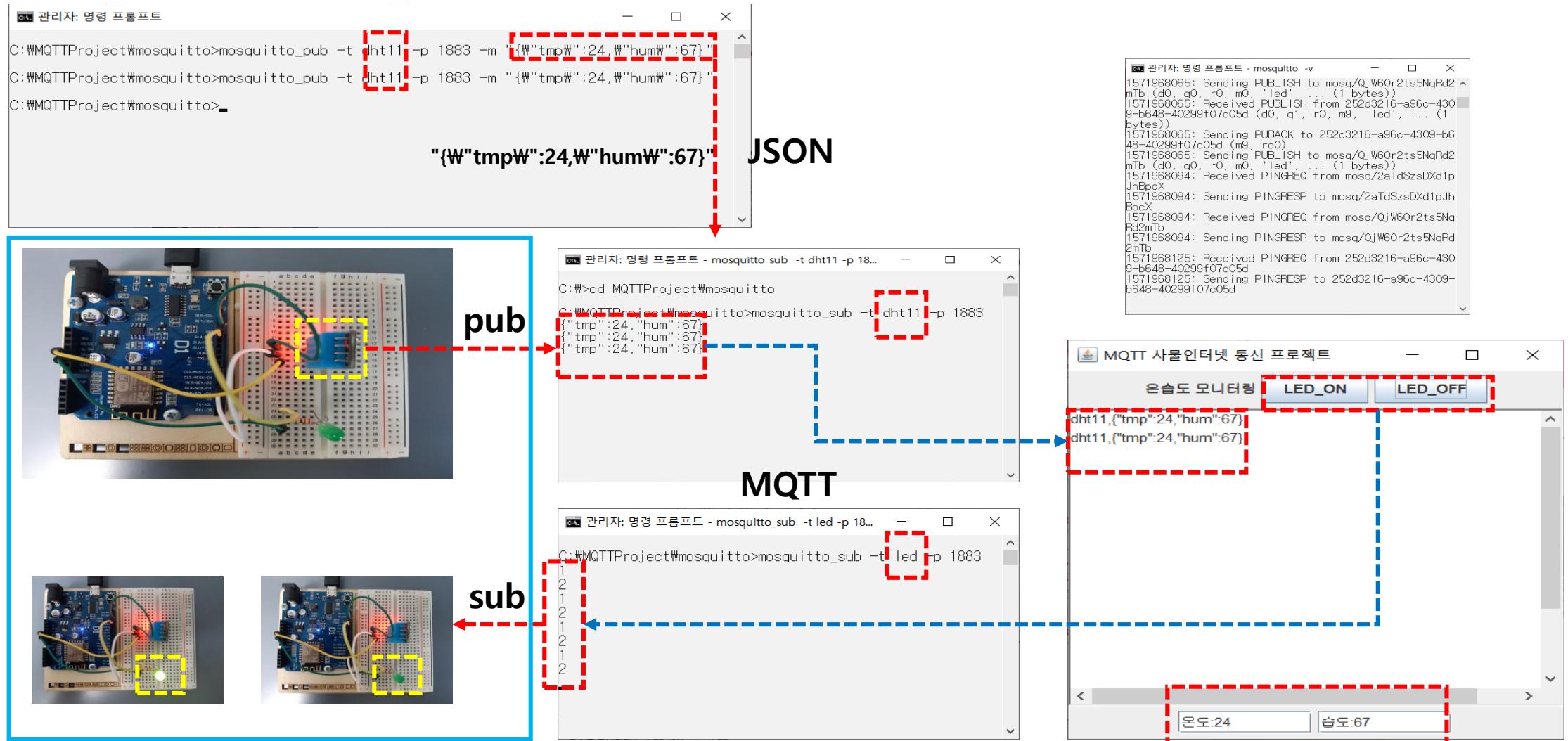
1. Java를 활용하여 DHT11 센서 데이터 모니터링 및 LED 제어하기

Java MQTT Client

WeMos D1



1. Java에서 MQTT 연동하기(subscribe, publish)



2. Source Code → kr.javatpc package에 만들자.

```

public class Project05_F {
    public class MqttClass implements MqttCallback {
        Runnable task1 = new Runnable(){
            @Override
            public void run() {
                try {
                    String clientId = UUID.randomUUID().toString();
                    //new MqttClient()
                    client = new MqttClient("tcp://172.30.1.15:1883", clientId);
                    MqttConnectOptions connopt = new MqttConnectOptions();
                    connopt.setCleanSession(true);
                    client.connect(connopt);
                    client.setCallback(MqttClass.this);
                    client.subscribe("dht11");
                    new IoTFrame(MqttClass.this);
                } catch (MqttException e) {
                    System.out.println("ERR0"+e.getStackTrace());
                }
            }
        };
        public void connectionLost(Throwable cause) {
        }

        public void deliveryComplete(IMqttDeliveryToken token) {
        }

        public void messageArrived(String topic, MqttMessage msg) {
            System.out.println("Topic : " + topic + " Message : " + new String(msg.getPayload()));
        }

        public void sendMessage() {
            client.publish("dht11", "Hello MQTT");
        }
    }
}

public interface ReceiveEventListner extends ActionListener {
}

```

自身의 IP로 설정할 것

mosquito_sub -t dht11

dht11 topic에 데이터가 수신되면

public void recvMsg(String topic, MqttMessage msg);

public interface ReceiveEventListner

public class IoTFrame extends JFrame implements ActionListener,ReceiveEventListner

3. Source Code

```

import java.util.UUID;
import org.eclipse.paho.client.mqttv3.*;
public class MqttClass implements MqttCallback{

    private MqttClient client = null;
    public MqttClass(){
        new Thread(task1).start();
    }
    private ReceiveEventListner listener = null;

    Runnable task1 = new Runnable(){
        @Override
        public void run() {
            try {
                String clientId = UUID.randomUUID().toString();
                //new MqttClient()
                client = new MqttClient("tcp://172.30.1.15:1883", clientId);
                MqttConnectOptions connopt = new MqttConnectOptions();
                connopt.setCleanSession(true);
                client.connect(connopt);
                client.setCallback(MqttClass.this);
                client.subscribe("dht11");

                new IoTFrame(MqttClass.this);

            } catch (MqttException e) {
                System.out.println("ERR0"+e.getStackTrace());
            }
        }
    };
}

```

```

public void sendMessage(String payload){
    MqttMessage message = new MqttMessage();
    message.setPayload(payload.getBytes());
    try {
        if(client.isConnected()){
            client.publish("led", message);
        }
    } catch (MqttException e) {
        System.out.println("error1-"+e.getStackTrace());//+e.getMessage());
    }
}

@Override
public void connectionLost(Throwable arg0) {
    try {
        System.out.println("disconect");
        client.close();
    } catch (MqttException e) {
        System.out.println("error"+e.getMessage());
    }
}
@Override
public void deliveryComplete(IMqttDeliveryToken arg0) {
}
public void setMyEventListner(ReceiveEventListner listener){
    this.listener = listener;
}
@Override
public void messageArrived(String topic, MqttMessage msg) throws Exception {
    //System.out.println(topic+","+msg.toString());
    listener.recvMsg(topic, msg);
}

```

PART-3

4. Source Code

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import org.eclipse.paho.client.mqttv3.MqttMessage;
import org.json.JSONObject;
public class IoTFrame extends JFrame implements
    ActionListener,ReceiveEventListner{
    private static final long serialVersionUID = 1L;
    private JTextField tmp = new JTextField(10);
    private JTextField hum = new JTextField(10);
    private JButton ledOn = new JButton("LED_ON");
    private JButton ledOff = new JButton("LED_OFF");
    private JLabel msg = new JLabel("온습도 모니터링");
    private JTextArea out = new JTextArea(20,40);
    private JPanel panel = new JPanel();
    private JPanel panel1 = new JPanel();
    private JPanel panel2 = new JPanel();
    private ScrollPane sp=new ScrollPane();
    private MqttClass mqtt = null;

    public IoTFrame(MqttClass mqtt){
        this();
        this.mqtt = mqtt;
        this.mqtt.setMyEventListner(this);
    }
}

```

```

public IoTFrame(){
    super("MQTT 사물인터넷 통신 프로젝트");
    setSize(400,400);
    panel.add(msg);
    panel.add(ledOn);
    panel.add(ledOff);
    panel1.add(tmp);
    panel1.add(hum);
    sp.add(out);
    add(BorderLayout.NORTH, panel);
    add(BorderLayout.CENTER, sp);
    add(BorderLayout.SOUTH, panel1);
    //add(BorderLayout.EAST, panel2);
    ledOn.addActionListener(this);
    ledOff.addActionListener(this);
    setVisible(true);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}

@Override
public void recvMsg(String topic, MqttMessage msg) {
    System.out.println(topic+", "+msg);
    String append = out.getText();
    out.setText(topic+", "+msg+"\n"+append);
    JSONObject obj=new JSONObject(new String(msg.getPayload()));
    tmp.setText("온도:"+obj.get("tmp").toString());
    hum.setText("습도:"+obj.get("hum").toString());
}

@Override
public void actionPerformed(ActionEvent e) {
    JButton b = (JButton) e.getSource();
    if(b.getText().equals("LED_ON")){
        mqtt.sendMessage("1");
    }else if(b.getText().equals("LED_OFF")) {
        mqtt.sendMessage("2");
    }
}
}

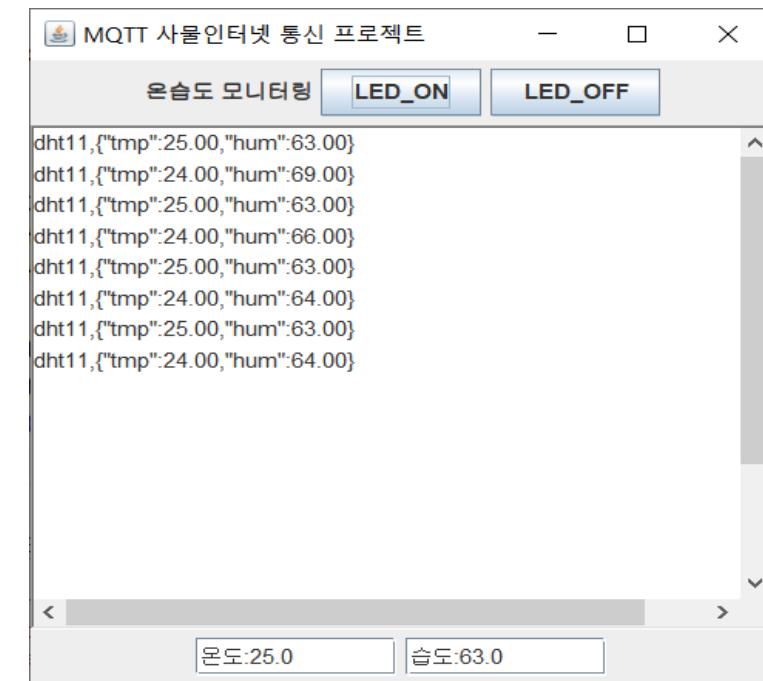
```



5. Source Code

```
import org.eclipse.paho.client.mqttv3.MqttMessage;
public interface ReceiveEventListner {
    public void recvMsg(String topic, MqttMessage msg);
}

public class JavaMQTT{
    public static void main(String[] args) {
        new MqttClass();
    }
}
```



9. Java MQTT Client(STET 2)

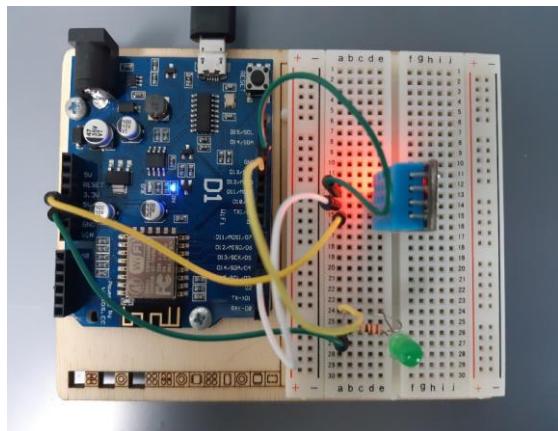
JavaThinkingPresentationCoding

PART-3

1. Java 시연하기

4

WeMos 구동



관리자: 명령 프롬프트 - mosquitto -v

```
C:\#MQTTProject#mosquitto -v
1569420036: mosquitto version 1.6.6 starting
1569420036: Using default
1569420036: Opening ipv6
1569420036: Opening ipv4
1569420036: New connection from ::1 on port 1883.
```

1

MQTT Server 구동

관리자: 명령 프롬프트 - mosquitto_sub -t dht11 -p 1883

```
{"tmp":25.00,"hum":68.00}
{"tmp":25.00,"hum":68.00}
{"tmp":25.00,"hum":68.00}
{"tmp":25.00,"hum":68.00}
{"tmp":25.00,"hum":73.00}
{"tmp":25.00,"hum":68.00}
```

2

dht11 sub 구동

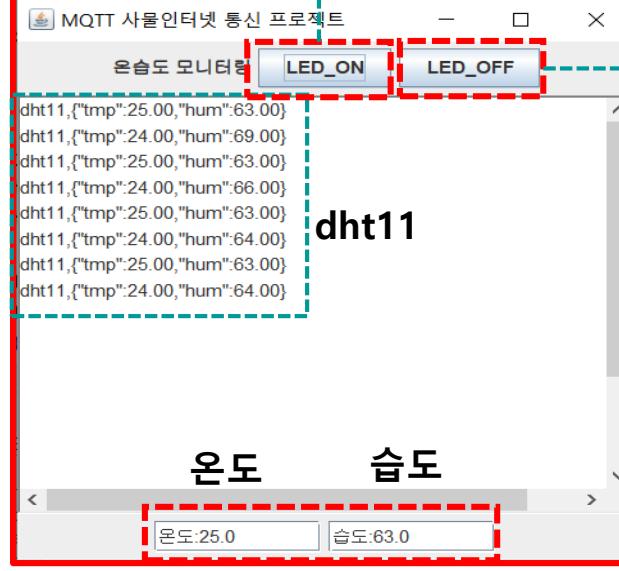
선택 관리자: 명령 프롬프트 - mosquitto_sub -t led -p 1883

```
C:\#Windows>cd..
C:\#>cd MQTTProject#mosquitto
C:\#MQTTProject#mosquitto_sub -t led -p 1883
1
1
1
1
2
1
2
1
2
1
2
```

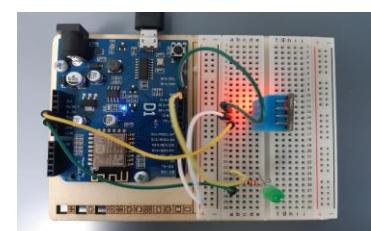
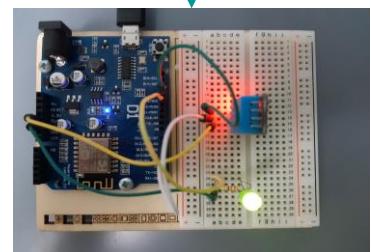
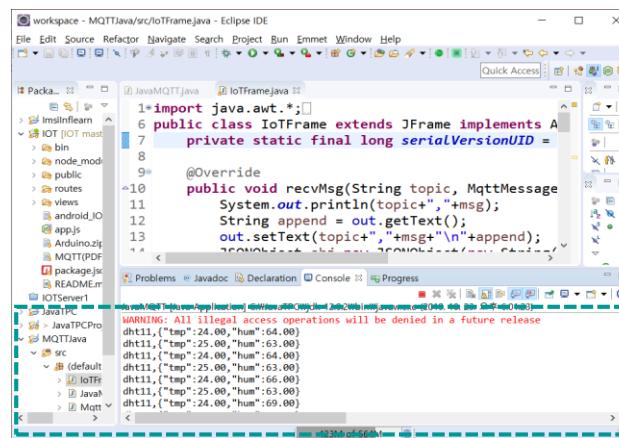
3

led sub 구동

5 Java 구동

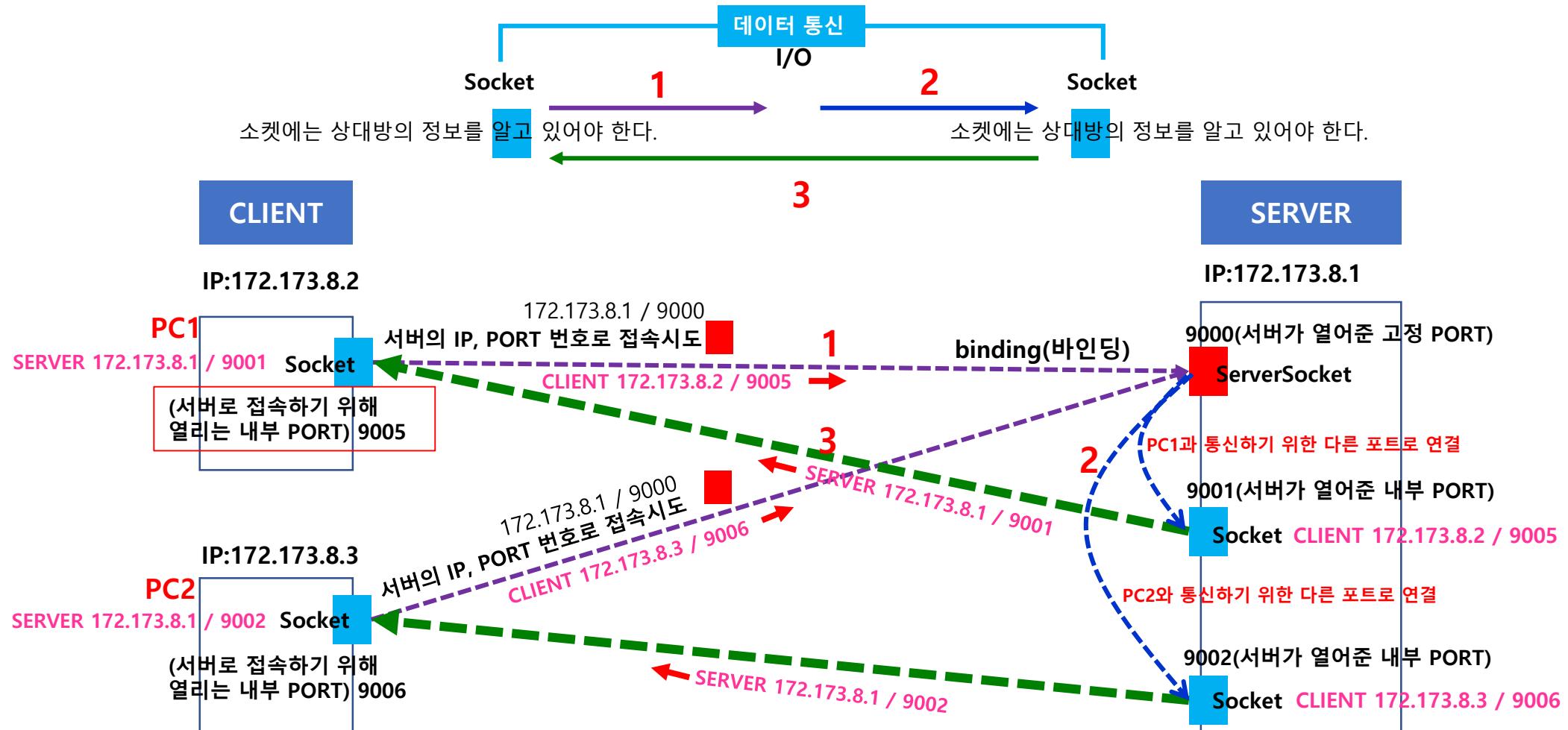


온도 습도

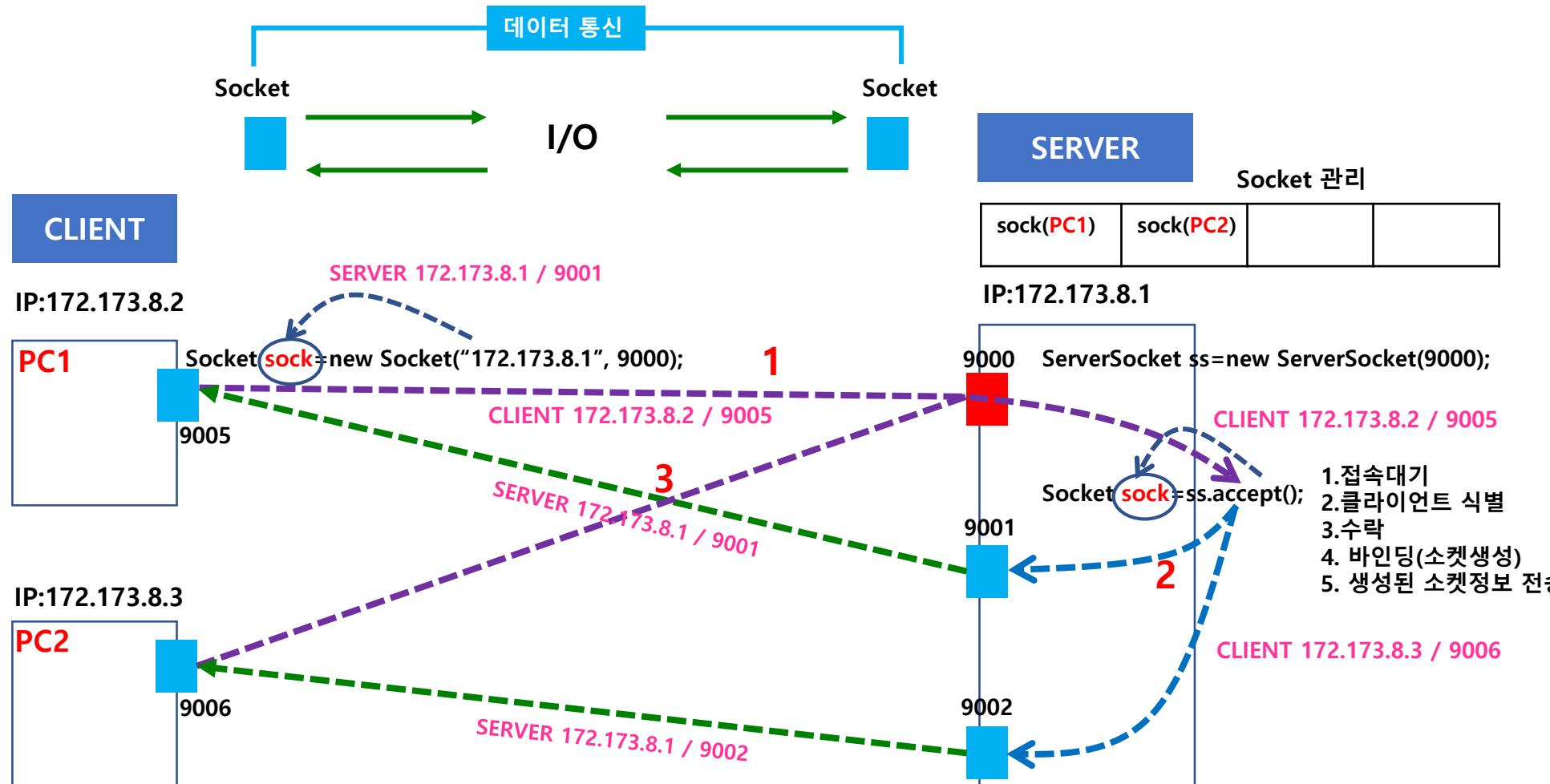


PROJECT 6 Java Socket(소켓) Multi-Chatting 프로그래밍

① Socket(소켓)이 만들어 지는 과정(TCP 3-way Handshake)



2. Java에서 Socket(소켓) 만들기(ServerSocket, Socket , import java.net.*)



2. Socket(소켓)을 이용한 ECHO 프로그램 만들기

CLIENT

```
import java.io.*;
import java.net.Socket;
import java.util.Scanner;

public class Project06A_Client {
    public static void main(String[] args) {
        try{
            Socket socket=new Socket("127.0.0.1", 9999);
            System.out.println("Connection Success!");
            Scanner scanner = new Scanner(System.in);
            String message = scanner.nextLine();
            OutputStream out = socket.getOutputStream();
            DataOutputStream dos = new DataOutputStream(out);
            dos.writeUTF(message);
            /////////////////////////////////
            InputStream in = socket.getInputStream();
            DataInputStream dis = new DataInputStream(in);
            System.out.println("Receive:"+dis.readUTF());
            dis.close();
            dos.close();
            socket.close();
        }catch(Exception e) {
            e.printStackTrace();
        }
    }
}//main
}//class
```

SERVER

```
import java.io.*;
import java.net.*;
public class Project06A_Server{
    public static void main(String[] args) {
        ServerSocket serverSocket = null;
        try {
            serverSocket = new ServerSocket(9999);
            System.out.println("server ready...");
        } catch(IOException e) {
            e.printStackTrace();
        }
        while(true) {
            try {
                Socket socket= serverSocket.accept();
                System.out.println("client connect success!");
                InputStream in = socket.getInputStream();
                DataInputStream dis = new DataInputStream(in);
                String message = dis.readUTF();
                OutputStream out = socket.getOutputStream();
                DataOutputStream dos = new DataOutputStream(out);
                dos.writeUTF("[ECHO]"+message+" (from Server!)");
                dos.close();
                dis.close();
                socket.close();
                System.out.println("client socket close...");
            } catch(IOException e) {
                e.printStackTrace();
            }
        }
    }
}//main
}//class
```

3. Socket(소켓)을 이용한 ECHO 프로그램 만들기(실행화면)

The screenshot shows two terminal windows side-by-side, labeled 'CLIENT' and 'SERVER'. Both windows have a title bar '관리자: 명령 프롬프트'.

CLIENT Window:

- Output:

```
C:\#JavaTCP\workspace\JavaTCPProject\bin>java Project06A_Client
Connection Success!
야호
Receive: [ECHO] 야호(from Server!)
Hello
Receive: [ECHO] Hello(from Server!)

C:\#JavaTCP\workspace\JavaTCPProject\bin>java Project06A_Client
Connection Success!
안녕하세요
Receive: [ECHO] 안녕하세요(from Server!)

C:\#JavaTCP\workspace\JavaTCPProject\bin>
```

SERVER Window:

- Output:

```
C:\#JavaTCP\workspace\JavaTCPProject\bin>java Project06A_Server
Server ready....
client connect success!
client socket close....
```

Diagram Labels:

- ECHO**: A vertical label positioned between the two terminal windows, indicating the flow of data between them.
- CLIENT**: A blue rectangular label above the left terminal window.
- SERVER**: A blue rectangular label above the right terminal window.

Dashed red arrows indicate the direction of data transmission between the two terminals, corresponding to the 'ECHO' label.

1. Socket(소켓)을 이용한 Multi-Chatting 만들기

```

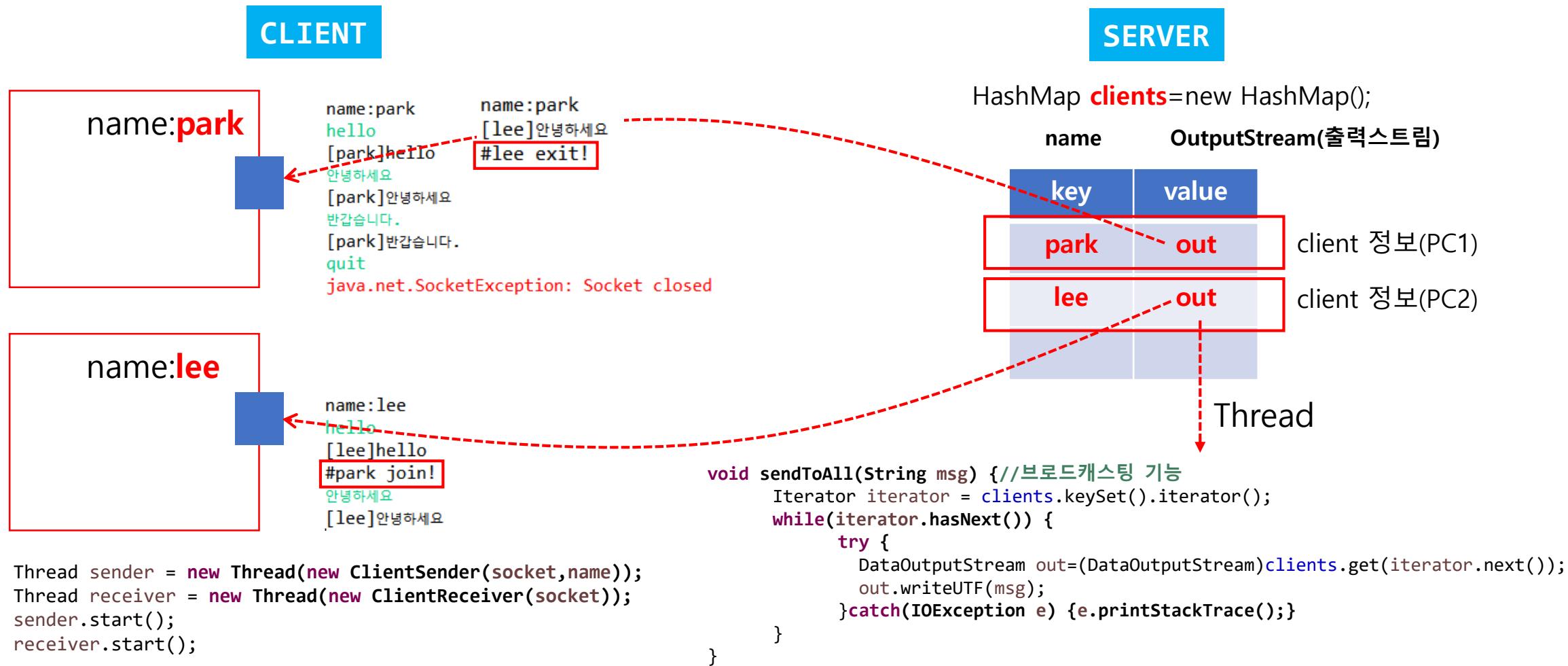
[SERVER]
C:\JavaTPC\workspace\JavaTCPProject\bin>java Project06F_MultiChatServer start server...
/127.0.0.1:63486 connect!
/127.0.0.1:63496 connect!

[CLIENT 63486]
관리자: 명령 프롬프트 - java Project06F_MultiChatClient
C:\JavaTPC\workspace\JavaTCPProject\bin>java Project06F_MultiChatClient name:park
 입장
#lee join!
hello_lee
[park]hello_lee
[lee]hello_park
안녕하세요
[park]안녕하세요
[lee]반갑습니다.
퇴장
quit
java.net.SocketException: Socket closed
    at java.base/java.net.SocketInputStream.socketRead0(Native Method)
    at java.base/java.net.SocketInputStream.socketRead(SocketInputStream.java:115)
    at java.base/java.net.SocketInputStream.read(SocketInputStream.java:168)
    at java.base/java.net.SocketInputStream.read(SocketInputStream.java:140)
    at java.base/java.net.SocketInputStream.read(SocketInputStream.java:200)
    at java.base/java.io.DataInputStream.readUnsignedShort(DataInputStream.java:342)
    at java.base/java.io.DataInputStream.readUTF(DataInputStream.java:594)
    at java.base/java.io.DataInputStream.readUTF(DataInputStream.java:569)
    at Project06F_MultiChatClient$ClientReceiver.run(Project06F_MultiChatClient.java:50)
    at java.base/java.lang.Thread.run(Thread.java:835)

[CLIENT 63496]
관리자: 명령 프롬프트 - java Project06F_MultiChatClient
C:\JavaTPC\workspace\JavaTCPProject\bin>java Project06F_MultiChatClient
name: lee
[park]hello_lee
hello_park
[lee]hello_park
[park]안녕하세요
반갑습니다.
[lee]반갑습니다.
#park exit!

```

2. Socket(소켓)을 이용한 Multi-Chatting 만들기



PART-3

3. Socket(소켓)을 이용한 Multi-Chatting 만들기

```

import java.io.*;
import java.net.*;
import java.util.*;
public class Project06F_MultiChatServer {
    HashMap clients;
    MultiChatServer() {
        clients = new HashMap();
        Collections.synchronizedMap(clients);
    }
    public void start() {
        ServerSocket serverSocket = null;
        Socket socket = null;
        try{
            serverSocket = new ServerSocket(9999);
            System.out.println("start server...");
            while(true) {
                socket = serverSocket.accept();
                System.out.println(socket.getInetAddress()+":"+socket.getPort()+" connect!");
                ServerReceiver thread = new ServerReceiver(socket);
                thread.start(); —————— Thread 실행
            }//while
        }catch(Exception e) {e.printStackTrace();}
    }
    void sendToAll(String msg) { //브로드캐스팅 기능
        Iterator iterator = clients.keySet().iterator();
        while(iterator.hasNext()) {
            try {
                DataOutputStream out =(DataOutputStream)clients.get(iterator.next());
                out.writeUTF(msg);
            }catch(IOException e) {e.printStackTrace();}
        }//while
    }
    public static void main(String[] args) {
        new Project06F_MultiChatServer .start();
    }
}

```

브로드캐스팅 기능

```

class ServerReceiver extends Thread {
    Socket socket; DataInputStream in; DataOutputStream out;
    ServerReceiver(Socket socket) {
        this.socket = socket;
    }
    try{
        in = new DataInputStream(socket.getInputStream());
        out = new DataOutputStream(socket.getOutputStream());
    }catch(Exception e) {e.printStackTrace();}
}
public void run() {
    String name = "";
    try{
        name = in.readUTF();
        if (clients.get(name) != null) { //같은 이름 사용자 존재
            out.writeUTF("#Already exist name : "+name);
            out.writeUTF("#Please reconnect by other name");
            System.out.println(socket.getInetAddress()+":"+socket.getPort()+" disconnect!");
            in.close();
            out.close();
            socket.close();
            socket = null;
        } else { //같은 이름 존재하지 않는 경우
            sendToAll("#"+name+" join!");
            clients.put(name, out);
        }
        while(in != null) { sendToAll(in.readUTF()); }
    }else
    }catch(IOException e) { e.printStackTrace();
    }finally{
        if (socket != null) {
            sendToAll("#"+name+" exit!");
            clients.remove(name);
            System.out.println(socket.getInetAddress()+":"+socket.getPort()+" disconnect!");
        }
    }finally
}
//inner class
}//class

```

4. Socket(소켓)을 이용한 Multi-Chatting 만들기

```
import java.io.*;
import java.net.Socket;
import java.util.Scanner;
public class Project06F_MultiChatClient {
public static void main(String[] args) {
try {
    Socket socket = new Socket("127.0.0.1", 7777);
    Scanner scanner = new Scanner(System.in);
    System.out.print("name:");
    String name = scanner.nextLine();
    Thread sender = new Thread(new ClientSender(socket, name));
    Thread receiver = new Thread(new ClientReceiver(socket));
    sender.start();
    receiver.start();
} catch(Exception e) { e.printStackTrace(); }
} //main
```

보내는 Thread

```
static class ClientSender extends Thread {
    Socket socket; DataOutputStream out; String name;
    ClientSender(Socket socket, String name) {
        this.socket = socket; this.name = name;
        try {
            out = new DataOutputStream(socket.getOutputStream());
        } catch(Exception e) { e.printStackTrace(); }
    } //생성자
    public void run() {
        Scanner scanner = new Scanner(System.in);
        try{
            if (out != null) out.writeUTF(name);
            while(out != null) {
                String message = scanner.nextLine();
                if (message.equals("quit")) break;
                out.writeUTF("[ "+name+" ]"+message);
            } //while
            out.close();
            socket.close();
        } catch(Exception e) { e.printStackTrace(); }
    } //run
} //inner_class
```

받는 Thread

```
static class ClientReceiver extends Thread {
    Socket socket; DataInputStream in;
    ClientReceiver(Socket socket) {
        this.socket = socket;
        try{
            in = new DataInputStream(socket.getInputStream());
        } catch(Exception e) { e.printStackTrace(); }
    } //생성자
    public void run() {
        while(in != null) {
            try{
                System.out.println(in.readUTF());
            } catch(Exception e) {
                e.printStackTrace();
                break;
            }
        } //while
        try {
            in.close();
            socket.close();
        } catch(Exception e) {
            e.printStackTrace();
        }
    } //run
} //inner_class
} //class
```

