**Naver Map Open API**

1. API 이용 신청
   1. <https://developers.naver.com/main/> 를 통해 Map Open API 어플리케이션 이용 신청
   2. 
      1. 웹서비스의 url은 아무렇게 써도 상관없음, 이후 나오는 Android 패키지 명도 아무렇게 사용해도 무관함. 이 후 정식 프로젝트에서 올바르게 적어야함

|  |
| --- |
| 1. using System; 2. using System.Collections.Generic; 3. using System.Linq; 4. using System.Text; 5. using System.Threading.Tasks; 6. using System.Net; 7. using System.IO; 8. namespace GoogleMapOpenAPI\_0425 9. { 10. class APIExamTranslate 11. { 12. public static string Start(String target) 13. { 14. string query = target; // 검색할 주소 15. //string url = "https://openapi.naver.com/v1/map/geocode?query=" + query; // 결과가 JSON 포맷 16. string url = "https://openapi.naver.com/v1/map/geocode.xml?query=" + query; // 결과가 XML 포맷 17. HttpWebRequest request = (HttpWebRequest)WebRequest.Create(url); 18. request.Headers.Add("X-Naver-Client-Id", "penrf9sjWqff0tZLTxbJ"); // 클라이언트 아이디 19. request.Headers.Add("X-Naver-Client-Secret", "At5SVJCTTo"); // 클라이언트 시크릿 20. HttpWebResponse response = (HttpWebResponse)request.GetResponse(); 21. string status = response.StatusCode.ToString(); 22. if (status == "OK") 23. { 24. Stream stream = response.GetResponseStream(); 25. StreamReader reader = new StreamReader(stream, Encoding.UTF8); 26. string text = reader.ReadToEnd(); 27. return text; 28. //Console.WriteLine(text); 29. } 30. else 31. { 32. return "Error 발생=" + status; 33. //Console.WriteLine("Error 발생=" + status); 34. } 35. } 36. } 37. } |

* 1. ("X-Naver-Client-Id 에 내 어플리케이션 에서 방금 만든 어플리케이션의 id, 시크릿 코드를 입력하면 해당 코드가 돌아간다.

2. XML 요소 분석 <쿼리문 = 동대전로>

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8" ?>  <result>  <userquery><![CDATA[동대전로]]></userquery>  <total>1</total>  <items>  <item>  <address><![CDATA[대전광역시 동구 동대전로]]></address>  <addrdetail>  <country><![CDATA[대한민국]]></country>  <sido><![CDATA[대전광역시]]></sido>  <sigugun><![CDATA[동구]]></sigugun>  <dongmyun><![CDATA[동대전로]]></dongmyun>  <ri><![CDATA[]]></ri>  <rest><![CDATA[]]></rest>  </addrdetail>  <isRoadAddress>true</isRoadAddress>  <point>  <x>127.4467826</x>  <y>36.3367507</y>  </point>  </item>  </items>  </result> |

RootNode : result

CildNode

1. Userquery : 유저가 보낸 쿼리문
2. Total : 검색된 총 item 수
3. Items : item Root node
   1. Item
      1. Address : 일반 주소
      2. Addrdetail : 세부 주소
         1. Country : 나라
         2. Sido : 시,도
         3. Sigugun : 구
         4. Dongmyun : 동명, 도로명
         5. Ri : ?????
         6. Rest : ?????
      3. isRoadAddress : 이 검색이 도로명 검색인가. Bool값
      4. point : 현재 이 주소의 좌표값 y 경도, x 위도

**NaverSearchOpenAPI**

1. 인식…위와 같음

2. 코드

|  |
| --- |
| string query = target; // 검색할 문자열  //string url = "https://openapi.naver.com/v1/search/blog?query=" + query; // 결과가 JSON 포맷  string url = "https://openapi.naver.com/v1/search/local.xml?query=" + query; // 결과가 XML 포맷  HttpWebRequest request = (HttpWebRequest)WebRequest.Create(url);  request.Headers.Add("X-Naver-Client-Id", "ONQTxiZPzHemKitDqIpe"); // 클라이언트 아이디  request.Headers.Add("X-Naver-Client-Secret", "3SNeTNy2Lt"); // 클라이언트 시크릿  HttpWebResponse response = (HttpWebResponse)request.GetResponse();  string status = response.StatusCode.ToString();  if (status == "OK")  {  Stream stream = response.GetResponseStream();  StreamReader reader = new StreamReader(stream, Encoding.UTF8);  string text = reader.ReadToEnd();  return text;  //Console.WriteLine(text);  }  else  {  Console.WriteLine("Error 발생=" + status);  return "Error 발생=" + status;  } |

1. xml분석

|  |
| --- |
| 1. <?xml version="1.0" encoding="UTF-8"?><rss version="2.0"> 2. <channel> 3. <title>Naver Open API - local ::'동대전로 음식점'</title> 4. <link>http://search.naver.com</link> 5. <description>Naver Search Result</description> 6. <lastBuildDate>Wed, 25 Apr 2018 13:45:17 +0900</lastBuildDate> 7. <total>208</total> 8. <start>1</start> 9. <display>10</display> 10. <item> 11. <title>박가네</title> 12. <link></link> 13. <category>한식&gt;돼지고기구이</category> 14. <description>매운양푼등갈비찜, 등갈비 석쇠구이 등 등갈비 전문점.</description> 15. <telephone>042-621-6219</telephone> 16. <address>대전광역시 동구 가양동 419-3</address> 17. <roadAddress>대전광역시 동구 동대전로 287</roadAddress> 18. <mapx>350357</mapx> 19. <mapy>416341</mapy> 20. </item> 21. 생략. |

Root : channel

Child node :

Title

Link

Description

lastbuildDate

total

start

display

item

title : 음식점 이름

link : 음식점 사이트

category : 음식점 종류

description : 음식점 설명

telephone : 음식점 전화번호

address : 음식점 일반 주소

roadaddress : 음식점 도로명 주소

mapx : ????

mapy : ?????

**주변 음식점 검색하는 프로그램**

필요한 요소

1. 현재 자기 자신의 위치 주소
2. 1번에 기반한 주변 10개의 음식점 주소

화면 중앙에 찾은 음식점 10개 위치 마커로 찍기

주변에는 xml로 찾은 정보를 텍스트로 출력