

크롤링

목차

3.1 웹 페이지 자료 추출

3.2 트위터 자료 추출

웹 페이지 자료 추출

- 크롤링(crawling) 또는 스크레이핑(scraping)은 웹 페이지를 그대로 가져와서 거기서 데이터를 추출해 내는 행위
 - 크롤링하는 소프트웨어는 크롤러(crawler)
- 웹문서(사이트)는 통상 텍스트와 이미지가 혼합되어 있는 HTML 형식으로 구성
- 비구조화된 웹문서 자료를 정형화된(구조화된) 형태로 변환하여 데이터베이스나 스프레드시트에 저장, 분석할 수 있도록 하는 것

- R에서 웹문서 가져오기(웹스크랩/웹크롤링)
 - 웹에 있는 데이터를 가져오는 단계
 - 요청 : GET과 POST 방식
 - 추출 및 저장
 - 관련 R 패키지
 - XML, RCurl, httr, rvest 등

- **rvest의 동작 순서**

- **html 문서 데이터 가져오기**
- **필요한 노드 선택하기**
- **노드에서 필요한 작업**
 - **노드에서 text를 가져오기**
 - **노드내에 특정 속성(attr)을 추출**

- **HTML 또는 XML 읽기(xml2 패키지)**

- **URL의 html 을 header 와 body로 가져옴**

```
> read_xml(x, encoding="")
```

- **x, : 문자열, 연결 또는 원시 벡터**
 - **문자열은 경로, URL 또는 리터럴 xml**
- **encoding : 문서의 기본 인코딩을 지정**
 - **XML문서는 UTF-8 또는 UTF-16으로 가정**

▪ 태그 찾기(rvest 패키지)

- xpath 및 css 선택기를 사용하여 HTML문서에서 조각을 추출

```
> html_nodes(x, css, xpath)
```

- x : 문서, 노드 세트 또는 단일 노드 중 하나
- css : 선택할 노드(태그)
 - css 또는 xpath 중 하나
- xpath : css 또는 xpath1.0 선택기를 사용
- 하위 태그를 사용시
 - html_nodes() %>% html_nodes()

▪ Xpath

- XPath(XML Path Language)는 W3C의 표준으로 확장 생성 언어 문서의 구조를 통해 경로 위에 지정한 구문을 사용하여 항목을 배치하고 처리하는 방법을 기술하는 언어
- XML 표현보다 더 쉽고 약어로 되어 있으며, XSL 변환(XSLT)과 XML 지시자 언어(XPointer)에 쓰이는 언어
- XPath는 XML 문서의 노드를 정의하기 위하여 경로식을 사용하며, 수학 함수와 기타 확장 가능한 표현들이 있음

```

<?xml version="1.0" encoding="utf-8"?>
<wikimedia>
  <projects>
    <project name="Wikipedia" launch="2001-01-05">
      <editions>
        <edition language="English">en.wikipedia.org</edition>
        <edition language="German">de.wikipedia.org</edition>
        <edition language="French">fr.wikipedia.org</edition>
        <edition language="Polish">pl.wikipedia.org</edition>
      </editions>
    </project>
    <project name="Wiktionary" launch="2002-12-12">
      <editions>
        <edition language="English">en.wiktionary.org</edition>
        <edition language="French">fr.wiktionary.org</edition>
        <edition language="Vietnamese">vi.wiktionary.org</edition>
        <edition language="Turkish">tr.wiktionary.org</edition>
      </editions>
    </project>
  </projects>
</wikimedia>

```

- **/wikimedia/projects/project/@name**
 - 위 XPath 식은 모든 project 요소의 name 속성을 선택
-
- **/wikimedia/projects/project/editions/edition[@language="English"]/text()**
 - 위 XPath 식은 모든 영문 Wikimedia 프로젝트의 주소(language 속성이 English인 모든 edition 요소의 문자열)를 선택
 - **/wikimedia/projects/project[@name="Wikipedia"]/editions/edition/text()**
 - 위 XPath 식은 모든 위키백과의 주소 (Wikipedia의 이름 특성을 가진 project 요소 아래에 존재하는 모든 edition 요소의 문자열)를 선택

- **문자 추출(rvest 패키지)**

- **HTML 문서에서 문자 추출**

```
> html_text(x, trim=FALSE)
```

- **x** : 문서, 노드 세트 또는 단일 노드 중 하나

- **trim** : 앞, 뒤 공백 제거(TRUE)

- **HTML 표(rvest 패키지)**

- **HTML 테이블을 데이터 프레임으로 처리**

```
> html_table(x, header=NA, trim=TRUE,  
+           fill=FALSE, dec=".")
```

- **HTML 표(rvest 패키지)**

- **HTML 테이블을 데이터 프레임으로 처리**

```
> html_table(x, header=NA, trim=TRUE, dec=".")
```

- **x** : 문서, 노드 세트 또는 단일 노드 중 하나

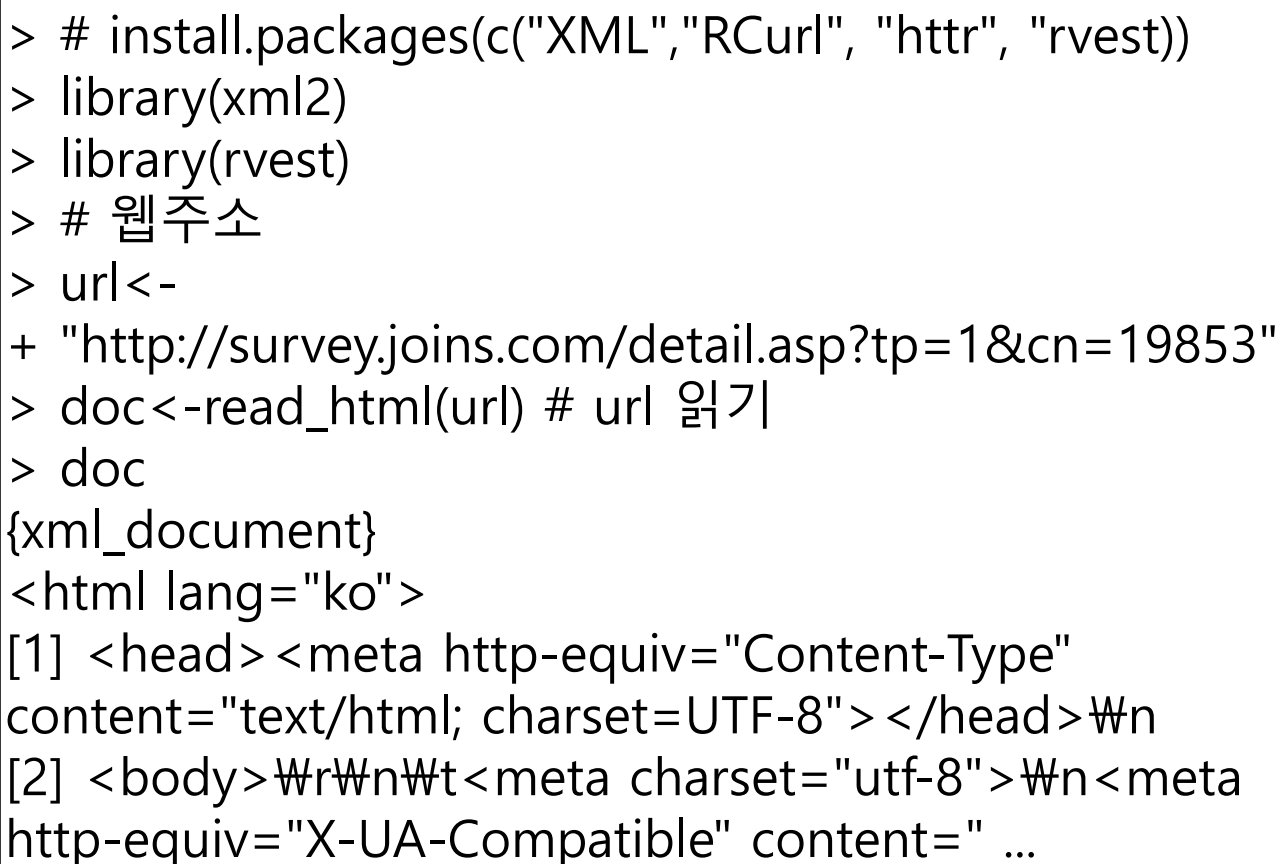
- **header** : 첫 번째 행의 변수명 사용 여부

- NA이면 <th> 태그를 변수명으로 사용

- **trim** : 앞, 뒤 공백 제거(TRUE)

- **dec** : 소수점 기호 문자

■ 다음 웹 페이지에서 조사개요 내용을 읽어오시오.
<http://survey.joins.com/detail.asp?tp=1&cn=19853>



```

> h.txt<-html_nodes(doc, 'div a') # <div> 태그의 내용
> h.txt
{xml_nodeset (5)}
[1] <a target="_blank"
href="http://article.joins.com/news/article/article.asp?total ...
[2] <a class="related_file"
href="javascript:KossdashowAttachFile('http://www.kosssda ...
a ...
[3] <a href="http://www.kosssda.or.kr/"
target="_blank">  [닫
기]</a>

```

```

> txt<-html_text(h.txt) # 해당 태그 텍스트 추출
> txt
[1] " 중앙일보 2014년 7월 16일자 5면" ""
[3] ""
[5] "[닫기]"
> h.txt<-html_nodes(doc, 'div') %>% html_nodes('a') #
<div> 태그의 하위 <a> 태그 의 내용
> # 또는 h.txt<-doc %>% html_nodes('div')
>
> txt<-html_text(h.txt) # 해당 태그 텍스트 추출
> txt
[1] " 중앙일보 2014년 7월 16일자 5면" ""
[3] ""
[5] "[닫기]"

```



```
> # <div> 태그내 style 옵션
> h.txt<-doc %>% html_nodes('div') %>%
+ html_nodes('[style]')
> h.txt
{xml_nodeset (4)}
[1] <div style="line-height:160%" align="justify"> 이
조사는 중앙일보가 2014년 7월 30일에 실시되는 국회 ...
[2] <div id="kossda_howto" style="position:relative;z-
index:99;display:none;">WrWnWt ...
[3] <div style="position:absolute;top:-
30px;left:350px;width:250px;height:50px;backg ...
[4] <span style="float:right"><a
href="javascript:show_kossda_howto('hide');">[닫
기]</ ...
```

```
> txt<-html_text(h.txt) # 해당 태그 텍스트 추출
> txt
[1] " 이 조사는 중앙일보가 2014년 7월 30일에 실시되는 국회의
원 재보궐선거의 판세를 예측하고자 출마 후보 확정 후 지역에
따라 1-2차례 시행한 것이다. 조사지역은 서울 동작구을, 대전 대
덕구, 경기 수원시을, 수원시병, 수원시정, 경기 김포시, 경기 평택
시을, 충북 충주시, 충남 서산시·태안군, 전남 순천시·곡성군 등 10
개 선거구이다. 이 자료는 서울 동작구을 선거구의 1차 조사에서
수집된 것으로 투표의사, 사전투표제도 이용의사, 지지후보, 지지
정당 등의 문항을 포함하고 있다. 자료에는 해당 지역의 성별 및
연령별 유권자수를 기준으로 산출한 가중치 변수가 포함되어 있
으므로 가중치를 부여한 상태에서 분석해야 한다. "
[2] "WrWnWtWtWtWtWtWtWtWtKOSSDA에서 [자료이용신청] 버튼을
선택하기 바랍니다.[닫기]WrWnWtWtWtWtWtWtWt"
[3] "KOSSDA에서 [자료이용신청] 버튼을선택하기 바랍니다.[닫
기]"
[4] "[닫기]"
```

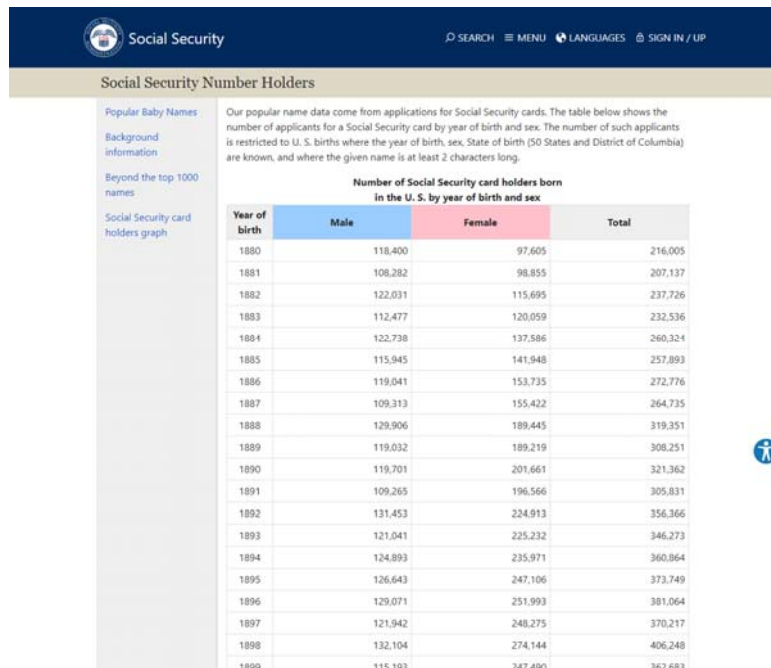
```
> # <div align="justify"> 태그
> h.txt<-doc %>% html_nodes('div') %>%
+ html_nodes('[align="justify"]')
> h.txt
{xml_nodeset (1)}
[1] <div style="line-height:160%" align="justify"> 이
조사는 중앙일보가 2014년 7월 30일에 실시되는 국회 ...
```

```
> txt<-html_text(h.txt) # 해당 태그 텍스트 추출
> txt
[1] " 이 조사는 중앙일보가 2014년 7월 30일에 실시되는
국회의원 재보궐선거의 판세를 예측하고자 출마 후보 확
정 후 지역에 따라 1-2차례 시행한 것이다. 조사지역은 서
울 동작구을, 대전 대덕구, 경기 수원시을, 수원시병, 수원
시정, 경기 김포시, 경기 평택시을, 충북 충주시, 충남 서산
시.태안군, 전남 순천시.곡성군 등 10개 선거구이다. 이 자
료는 서울 동작구을 선거구의 1차 조사에서 수집된 것으
로 투표의사, 사전투표제도 이용의사, 지지후보, 지지정당
등의 문항을 포함하고 있다. 자료에는 해당 지역의 성별
및 연령별 유권자수를 기준으로 산출한 가중치 변수가 포
함되어 있으므로 가중치를 부여한 상태에서 분석해야 한
다. "
> detach("package:rvest", unload=TRUE)
> detach("package:xml2", unload=TRUE)
```

예제 4.2

- 다음 웹 페이지의 표내용을 데이터 프레임으로 읽어 오시오.

<https://www.ssa.gov/oact/babynames/numberUSbirths.html>



The screenshot shows the 'Social Security Number Holders' page. It features a table titled 'Number of Social Security card holders born in the U. S. by year of birth and sex'. The table has four columns: 'Year of birth', 'Male', 'Female', and 'Total'. The data spans from 1880 to 1899. A sidebar on the left contains links for 'Popular Baby Names', 'Background information', 'Beyond the top 1000 names', and 'Social Security card holders graph'. A small blue icon is visible on the right side of the table.

Year of birth	Male	Female	Total
1880	118,400	97,605	216,005
1881	108,282	98,855	207,137
1882	122,031	115,695	237,726
1883	112,477	120,059	232,536
1884	122,738	137,586	260,324
1885	115,945	141,948	257,893
1886	119,041	153,735	272,776
1887	109,313	155,422	264,735
1888	129,906	189,445	319,351
1889	119,032	189,219	308,251
1890	119,701	201,661	321,362
1891	109,265	196,566	305,831
1892	131,453	224,913	356,366
1893	121,041	225,232	346,273
1894	124,893	235,971	360,864
1895	126,643	247,106	373,749
1896	129,071	251,993	381,064
1897	121,942	248,275	370,217
1898	132,104	274,144	406,248
1899	115,193	247,490	362,683

```
> library(xml2)
> library(rvest)
>
> url<-"https://www.ssa.gov/oact/babynames/
+ numberUSbirths.html"
> htxt<-read_html(url)      # html 코드 불러오기
> url.table<-html_table(html_nodes(htxt,
+                               "table")[[2]])
```

```

> head(url.table)
  Year ofbirth   Male Female  Total
1      1880 118,400  97,605 216,005
2      1881 108,282  98,855 207,137
3      1882 122,031 115,695 237,726
4      1883 112,477 120,059 232,536
5      1884 122,738 137,586 260,324
6      1885 115,945 141,948 257,893
>
> detach("package:rvest", unload=TRUE)
> detach("package:xml2", unload=TRUE)

```

▪ HTML 또는 XML 읽기(RCurl 패키지)

– URL의 html 가져옴

```
> getURL(x, encoding="")
```

– x, : 문자열, 연결 또는 원시 벡터

- 문자열은 경로, URL 또는 리터럴 xml

– encoding : 문서의 기본 인코딩을 지정

- 문자열은 'UTF-8' 또는 'ISO-8859-1'
- 정수는 CE_UTF8 및 CE_LAN1로 상징적으로 지정

■ 태그 찾기(XML 패키지)

- XML 또는 HTML 파일 또는 XML/XML내용이 들어 있는 문자열을 구문 분석하고 XML/XML 트리를 나타내는 R구조를 생성

> htmlParse(file)

- file : XML내용을 포함하는 파일의 이름

■ XML트리에서 일치하는 노드 찾기(XML 패키지)

> xpathSApply(doc, path, fun=NULL)

- doc : XML내부 문서의 개체
- path : XPath 문자열
- fun : 노드에 대한 처리할 때 사용되는 함수 개체 또는 식

예제 4.3

- 다음 웹 페이지에서 조사개요 내용을 읽어오시오.
<http://surveyjoins.com/detail.asp?tp=1&cn=19853>

The screenshot shows the '여론조사' (Public Opinion Survey) page on the 'Survey Joins' website. A red box highlights the '조사개요' (Survey Overview) section, which contains a table with the following information:

구분	내용
조사명	중앙일보 7.20 지방선거 여론조사, 2014 서울 동작구출, 1차
조사기관	중앙일보
주최	선거관리위원회(주최), 중앙일보(주최), 여론조사연구소(주최), 중앙일보(주최)
조사기간	2014.07.14 ~ 2014.07.15
조사대상	서울 동작구출 선거구
조사방법	면담조사
표본크기	1000(표본오차: ±3.1%p)
조사결과	중앙일보 2014년 7월 19일 발표

The page also includes a 'NEWSROOM' section with various news articles and a '조사개요' section with a list of survey details.

```

> library(XML)
> library(RCurl)
> url<-"http://survey.joins.com/detail.asp?tp=1&
+ cn=19853"
>
> doc<-getURL(url, .encoding="UTF-8")
> h.txt<-htmlParse(doc)
> txt<-xpathSApply(h.txt, path='//*[@id="body"]',
+                  xmlValue) # id='body' 내용
> txt<-gsub("[\n\t\r;]", " ", txt)

```

```

> txt
[1] "
          조사 개요          구분          내
용          조사명          중앙일보 7.30 재보궐선거 여론조사, 2014 : 서울 동
작구을, 1차          조사기관          중앙일보          키워드          선거여
론조사 보궐선거 총선 국회의원선거 투표의사 정당지지 지지정당
초록          이 조사는 중앙일보가 2014년 7월 30일에 실시되는 국회의원 재
보궐선거의 판세를 예측하고자 출마 후보 확정 후 지역에 따라 1-2차례 시행
한 것이다. 조사지역은 서울 동작구을, 대전 대덕구, 경기 수원시을, 수원시병,
수원시정, 경기 김포시, 경기 평택시을, 충북 충주시, 충남 서산시·태안군, 전남
순천시·곡성군 등 10개 선거구이다. 이 자료는 서울 동작구을 선거구의 1차 조
사에서 수집된 것으로 투표의사, 사전투표제도 이용의사, 지지후보, 지지정당
등의 문항을 포함하고 있다. 자료에는 해당 지역의 성별 및 연령별 유권자수를
기준으로 산출한 가중치 변수가 포함되어 있으므로 가중치를 부여한 상태에서
분석해야 한다.          조사지역          서울 동작구을 선거구          조사
대상자          서울 동작구을 선거구에 거주하는 만 19세 이상의 성인 남녀
조사기간          2014.07.14 - 2014.07.15          조사방법          전화면접
표본크기          800 (유효사례)          관련기사          중앙일보 2014년
7월 16일자 5면          관련파일          자료이용신청
KOSSDA에서 [자료이용신청] 버튼을선택하기 바랍니다.[닫기]
"

```

```
> # <div style="line-height:160%"> 태그 내용
> txt<-xpathSApply(h.txt, path='//*/div[@style="line-
+ height:160%"]', xmlValue)
> txt
[1] " 이 조사는 중앙일보가 2014년 7월 30일에 실시되는
국회의원 재보궐선거의 판세를 예측하고자 출마 후보 확
정 후 지역에 따라 1-2차례 시행한 것이다. 조사지역은 서
울 동작구을, 대전 대덕구, 경기 수원시을, 수원시병, 수원
시정, 경기 김포시, 경기 평택시을, 충북 충주시, 충남 서산
시.태안군, 전남 순천시.곡성군 등 10개 선거구이다. 이 자
료는 서울 동작구을 선거구의 1차 조사에서 수집된 것으
로 투표의사, 사전투표제도 이용의사, 지지후보, 지지정당
등의 문항을 포함하고 있다. 자료에는 해당 지역의 성별
및 연령별 유권자수를 기준으로 산출한 가중치 변수가 포
함되어 있으므로 가중치를 부여한 상태에서 분석해야 한
다. "
```

```
> # 첫 번째 테이블의 2행 1열
> txt<-xpathApply(h.txt,
+ path='//*/table[1]/tbody/tr[2]/td[1]', xmlValue)
> txt
[[1]]
[1] "중앙일보"

> detach("package:XML", unload=TRUE)
> detach("package:RCurl", unload=TRUE)
```

민원참여
시장경제의 확산, 자유롭고 공정한 경쟁 촉진

민원참여

- 상과법정 안내(연도)
- 성남**
- 상담안내/일자
- 비용상담지역
- 확장(수화)상담
- 주요 상담사례(Q&A)
- 신고서식
- 공정거래서비스신청
- 면담사무현황
- 공정위에 신고하기
- 국민신문고에 신고하기

주요상담사례(Q&A)

안전한 거래절차를 확립하고 불공정거래행위 등에 민원관련 주요 상담사례(Q&A)를 마련하였습니다.
이와 관련하여 문의사항이 있으시면 공정거래위원회 고객지원담당관실로 문의하시기 바랍니다.
1670-0007

* 첫오일/월제 단어를 입력후 검색버튼을 눌러주세요.

• 대표특목 공정거래정책 일반 +

• 소분특목 공정거래법 제도에 일반 +

번호	제목	등록부서	등록일	조회수
7	공정거래위원회 영장 표기	고객지원담당관실	2018-06-28	1267
6	공정거래법이 만들어진 경위(연혁)	경쟁정책과	2011-09-29	1245
5	공정거래법의 유래와 발전역사	경쟁정책과	2011-09-29	9301
4	외국의 공정거래위원회의 대차에 알고싶어요	경쟁정책과	2011-09-29	1054
3	공정거래위원회의 시장조치 내용과 효과	경쟁정책과	2011-09-29	2341
2	공정거래위원회의 소관 법률 및 주요 업무는 무엇입니까?	경쟁정책과	2011-09-29	3559
1	포상금 지급대상과 관련 규정	경쟁정책과	2011-09-30	4993

[▶](#)
[▶](#)
[▶](#)
[▶](#)
[▶](#)
[▶](#)
[▶](#)
[▶](#)

개인정보처리방침 |
 저작권정책 |
 국가보훈제도 |
 자유한국당 |
 RSS

©2018년 세종특별자치시 시청(국립 5호) 및 공정거래위원회
 설립인자 : 1670-0007 (공정거래위원회 콜센터)나 국민신문고에 인터넷 상으로 등록하여 운영함
 FAX : 044-200-4262 / 충청남도청내 부속기관
 Copyright © 2017 FAIR TRADE COMMISSION. All Rights Reserved.


```

> library(xml2)
> library(rvest)
> url<-
"http://www.ftc.go.kr/www/qnaDetailList.do?key=304&category=01&categorydetail=02"
> doc<-read_html(url) # url 읽기 <head>, <body>
> # <table> 태그의 2열과 3열, 5열 읽기
> html_table(doc)[[1]][, c(2,3,5)]

```

	제목	등록부서	조회수
1	경쟁제한적 규제란 무엇이며, 어떻게 추진되고 있는가?	규제개혁법무담당관	1207

```

> # <table> 태그 내용을 데이터 프레임으로 읽기
> res.table<-html_table(doc)

```

```

> res.table
[[1]]
  번호                      제목
1   1 경쟁제한적 규제란 무엇이며, 어떻게 추진되고 있는가?
      등록부서      등록일      조회수
1 규제개혁법무담당관 2011-09-26  1207

# 한글이 깨지는 경우 실행(아래와 같은 에러)
# Error in utils::type.convert(out[, i], as.is = TRUE, dec = dec) :
# invalid multibyte string at '<eb><91><90>?<b0>'
# UTF-8에서 인코딩된 것이 문제
# 한국어 OS를 사용하는 경우 대부분이 위와 같은 결과가 출력
# 이를 해결하는 방법은 Locale을 다음과 같이 설정하는 것
# Sys.getlocale()
# Sys.setlocale("LC_ALL", "English")

```

```

> Sys.setlocale("LC_ALL", "English")
[1] "LC_COLLATE=English_United
States.1252;LC_CTYPE=English_United
States.1252;LC_MONETARY=English_United
States.1252;LC_NUMERIC=C;LC_TIME=English_United States.1252"
> # 소분류목록 모두 읽기
> tot.table<-NULL
> for(i in 1:4) {
+   url<-
paste0("http://www.ftc.go.kr/www/qnaDetailList.do?key=304&cate
gory=07&categorydetail=0",i)
+   doc<-read_html(url)
+   # <table> 태그 내용을 데이터 프레임으로 읽기
+   res.table<-html_table(doc)
+   tot.table<-rbind(tot.table, res.table[[1]])
+ }
> Sys.setlocale("LC_ALL", "Korean")

```

```

> head(tot.table)
번호                      제목
1   11                      거래개시 거절
2   10              부당한 거래요구 거절후 강제 거래중단
3    9      입찰시 매각부지를 매입조건으로 시공하는조건부 업체 선정가능 여부
4    8      한업체에만 공급하는 제품에대해 타업체의 공급요구 거절한경우
5    7      제조회사의 일방적인 상품 공급 중단
6    6  6 제품을 대리점에 강매 후 외상매입금을 변제치 아니하는 이유로 공급을 중단
   등록부서 등록일   조회수
1      NA 2011-09-02   1118
2      NA 2011-09-02   1464
3      NA 2011-09-02    574
4      NA 2011-09-02   9999
5      NA 2011-09-02   5282
6      NA 2011-09-02   9943

> detach("package:rvest", unload=TRUE)
> detach("package:xml2", unload=TRUE)

```



```

> library(xml2)
Warning message:
패키지 'xml2'는 R 버전 3.4.4에서 작성되었습니다
> library(rvest)
Warning message:
패키지 'rvest'는 R 버전 3.4.4에서 작성되었습니다

# 한글이 깨지는 경우 실행(아래와 같은 에러)
# Error in utils::type.convert(out[, i], as.is = TRUE, dec = dec) :
# invalid multibyte string at '<eb><91><90>?<b0>'
> Sys.setlocale("LC_ALL", "English")
[1] "LC_COLLATE=English_United
States.1252;LC_CTYPE=English_United
States.1252;LC_MONETARY=English_United
States.1252;LC_NUMERIC=C;LC_TIME=English_United States.1252"

```

```

> # 1페이지~20페이지 까지 읽기
> tot.table<-NULL
> for(i in 1:20) {
+   url<-
paste0("http://www.mpm.go.kr/mpm/comm/newsInnoNotice/?mo
de=list&boardId=bbs_00000000000000020&pageldx=",i,"&searchC
ondition=all&searchKeyword=+&pageldx=1")
+   doc<-read_html(url)
+   res.table<-html_table(doc) # <table> 태그 내용을 데이터 프
레이밍으로 읽기
+   tot.table<-rbind(tot.table, res.table[[1]])
+ }
> Sys.setlocale("LC_ALL", "Korean")
[1]
"LC_COLLATE=Korean_Korea.949;LC_CTYPE=Korean_Korea.949;LC_
MONETARY=Korean_Korea.949;LC_NUMERIC=C;LC_TIME=Korean_K
orea.949"

```

```

> # 작성자 원도표 작성
> class(tot.table[, "작성일"]) <- "character"
> # 작성일 날짜 자료로 변환
> tot.table[, "작성일"] <- as.Date(tot.table[, "작성일"], format="%Y-%m-%d")
> # 작성일의 월 추출
> month <- as.numeric(format(tot.table[, "작성일"], "%m"))
> # 작성월이 6월 또는 7월에 작성한 작성자
> cus <- tot.table[(month %in% c(6, 7)), "작성자"]
> cus.tbl <- table(cus) # 작성자 빈도표
> cus.tbl

```

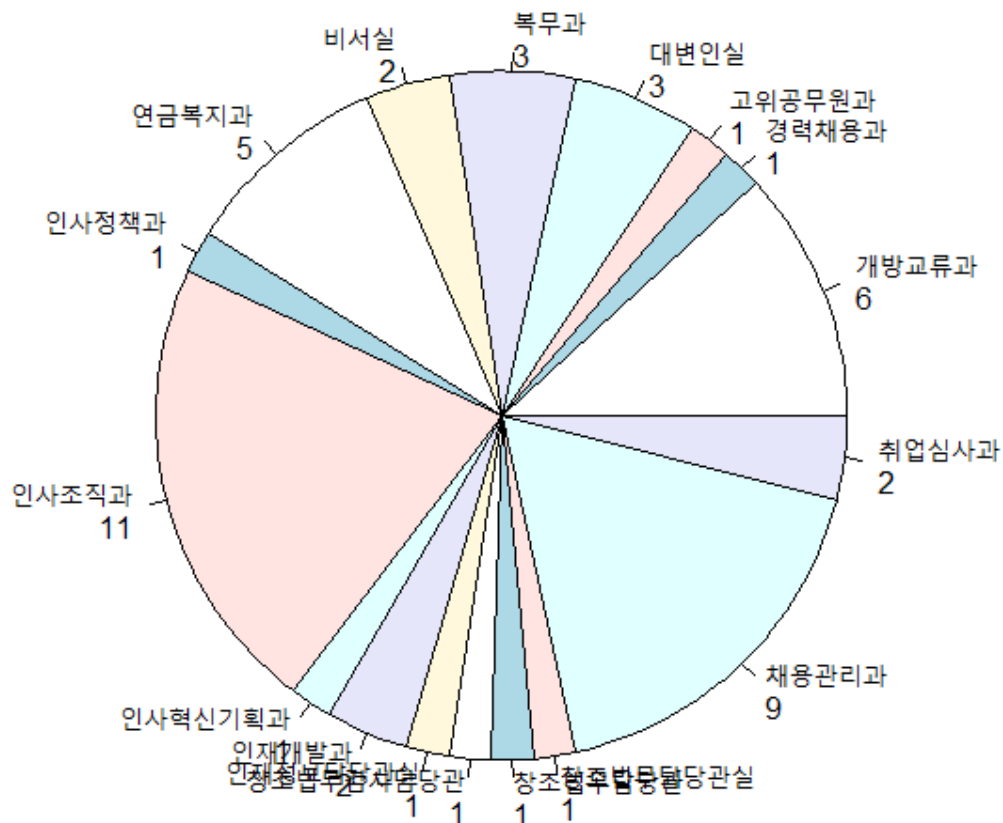
cus

개방교류과	경력채용과	고위공무원과	대변인실
6	1	1	3
복무과	비서실	연금복지과	인사정책과
3	2	5	1
인사조직과	인사혁신기획과	인재개발과	인재정보담당관실
11	1	2	1
창조법무감사담당관	창조법무담당관	창조법무담당관실	채용관리과
1	1	1	9
취업심사과			
2			

```

> windows()
> pie(cus.tbl, labels=paste(names(cus.tbl), cus.tbl, sep="₩n"))

```



```
> # 제목 내용 워드 클라우드
> library(KoNLP)
Checking user defined dictionary!
```

Warning message:

패키지 'KoNLP'는 R 버전 3.4.3에서 작성되었습니다

```
> library(stringr)
```

Warning message:

패키지 'stringr'는 R 버전 3.4.3에서 작성되었습니다

```
> useSejongDic()
```

Backup was just finished!

370957 words dictionary was built.

```
>
```

```
> txt.Data<-tot.table[, "제목"]
```

```
> # 명사 추출
```

```
> # Map은 주어진 벡터의 해당 요소에 함수를 적용
```

```
> wd<-Map(extractNoun, txt.Data)
```

```
> head(wd, n=2)
```

\$`2018년 인사혁신처 경력경쟁채용시험 서류전형 합격자
및 면접시험 일정 공고`

```
[1] "2018" "년" "인사" "혁신" "처" "경력" "경쟁"
```

```
[8] "채용시험" "서류" "전형" "합격자" "면접시험"
```

```
[13] "일정" "공"
```

\$`2018년도 국제기구 고용휴직 후보자 선발공고`

```
[1] "2018" "년" "국제기구" "고용" "휴직" "후보자"
```

```
[7] "선발" "공"
```

```
> # 중복 리스트 제거
> New.ls<-wd
> head(New.ls)
$`2018년 인사혁신처 경력경쟁채용시험 서류전형
합격자 및 면접시험 일정 공고`
[1] "2018" "년" "인사" "혁신" "처" "경력" "경쟁"
[8] "채용시험" "서류" "전형" "합격자" "면접시험"
[13] "일정" "공"

$`2018년도 국제기구 고용휴직 후보자 선발공고`
[1] "2018" "년" "국제기구" "고용" "휴직"
[6] "후보자" "선발" "공"
```

```
> # 리스트 성분 내에 중복 데이터 제거
> # 리스트 각 성분에 대한 함수 결과를 리스트
> # 돌려줌
> New.wd<-lapply(New.ls, unique)
> head(New.wd, n=2)
$`2018년 인사혁신처 경력경쟁채용시험 서류전형
합격자 및 면접시험 일정 공고`
[1] "2018" "년" "인사" "혁신" "처" "경력" "경쟁"
[8] "채용시험" "서류" "전형" "합격자" "면접시험"
[13] "일정" "공"

$`2018년도 국제기구 고용휴직 후보자 선발공고`
[1] "2018" "년" "국제기구" "고용" "휴직"
[6] "후보자" "선발" "공"
```

```

> # 숫자 제거
> clr.wd<-lapply(New.wd,
+               function(x) gsub("[[:digit:]]", "", x))
> # 길이가 2~10 사이의 단어 필터링 함수 정의
> filter1<-function(x){
+   (nchar(x)<=10 && nchar(x)>=2)
+ }
>
> filter2<-function(x){
+   Filter(filter1, x)
+ }

```

```

> # 줄 단어 대상 글자가 2글자~10글자 필터링
> lword<-sapply(clr.wd, filter2)
> head(lword, n=2)
$`2018년 인사혁신처 경력경쟁채용시험 서류전형
합격자 및 면접시험 일정 공고`
[1] "인사" "혁신" "경력" "경쟁" "채용시험" "서류"
[7] "전형"   "합격자"   "면접시험" "일정"

$`2018년도 국제기구 고용휴직 후보자 선발공고`
[1] "국제기구" "고용"   "휴직"   "후보자" "선발"

> TotalWD<-unlist(lword) # 리스트를 벡터로 변환

```



```
> detach("package:KoNLP", unload=TRUE)
> detach("package:stringr", unload=TRUE)
>
> detach("package:rvest", unload=TRUE)
> detach("package:xml2", unload=TRUE)
```

트위터 자료 추출

- 접근승인을 위한 키발급

<https://apps.twitter.com/app/new>

– 로그인

■ 접근승인을 위한 키발급 절차

Create an application

Application Details

Name *

Your application name. This is used to attribute the source of a tweet and in user-facing authorization screens. 32 characters max.

Description *

Your application description, which will be shown in user-facing authorization screens. Between 10 and 200 characters max.

Website *

Your application's publicly accessible home page, where users can go to download, make use of, or find out more information about your application. This fully-qualified URL is used in the source attribution for tweets created by your application and will be shown in user-facing authorization screens.
(If you don't have a URL yet, just put a placeholder here but remember to change it later.)

Callback URLs

Where should we return after successfully authenticating? [OAuth 1.0a](#) applications must explicitly specify their `oauth_callback` URL(s) here, as well as include the one of the URLs below in the request token step. To restrict your application from using callbacks, leave this field blank.

Add a Callback URL

Developer Agreement

☐ Yes, I have read and agree to the [Twitter Developer Agreement](#).

Create your Twitter application

Create an application

Application Details

Name *

Your application name. This is used to attribute the source of a tweet and in user-facing authorization screens. 32 characters max.

Description *

Your application description, which will be shown in user-facing authorization screens. Between 10 and 200 characters max.

Website *

Your application's publicly accessible home page, where users can go to download, make use of, or find out more information about your application. This fully-qualified URL is used in the source attribution for tweets created by your application and will be shown in user-facing authorization screens.
(If you don't have a URL yet, just put a placeholder here but remember to change it later.)

Callback URLs

Where should we return after successfully authenticating? [OAuth 1.0a](#) applications must explicitly specify their `oauth_callback` URL(s) here, as well as include the one of the URLs below in the request token step. To restrict your application from using callbacks, leave this field blank.

Add a Callback URL

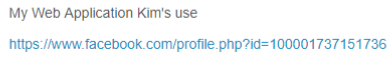
Developer Agreement

☒ Yes, I have read and agree to the [Twitter Developer Agreement](#).

Create your Twitter application

Test OAuth


Details Settings Keys and Access Tokens Permissions



Information about the organization or company associated with your application. This information is optional.

Organization	None
Organization website	None

Your application's Consumer Key and Secret are used to **authenticate** requests to the Twitter Platform.

Access level	Read and write (modify app permissions)
Consumer Key (API Key)	 manage keys and access tokens
Callback URL	https://www.facebook.com/profile.php?id=100001737151736
Callback URL Locked	Yes
Sign in with Twitter	Yes
App-only authentication	https://api.twitter.com/oauth2/token
Request token URL	https://api.twitter.com/oauth/request_token
Authorize URL	https://api.twitter.com/oauth/authorize
Access token URL	https://api.twitter.com/oauth/access_token

Delete Application

Test OAuth

Details Settings Keys and Access Tokens Permissions

Keep the "Consumer Secret" a secret. This key should never be human-readable in your application.

Consumer Key (API Key)	[REDACTED]
Consumer Secret (API Secret)	[REDACTED]
Access Level	Read and write (modify app permissions)
Owner	@lucy0nline@gmail.com
Owner ID	1098765432109876543210

[Regenerate Consumer Key and Secret](#) [Change App Permissions](#)

You haven't authorized this application for your own account yet.

By creating your access token here, you will have everything you need to make API calls right away. The access token generated will be assigned your application's current permission level.

Create my access token

▪ **문자열 기반으로 Twitter 검색 실행(twitteR 패키지)**

```
> searchTwitter(searchString, n=25, lang=NULL,  
+               since=NULL, until=NULL)
```

- **searchString**, : 검색 문자열
- **n=25**, : 반환할 최대 트윗
- **lang** : NULL이 아닌 경우 ISO639-1 코드로 제공되는 특정 언어로 트윗 제한
- **since** : 지정된 날짜 이후의 트윗 제한 (YYYY-MM-DD)
- **until** : 지정된 날짜까지 트윗을 허용 (YYYY-MM-DD)

▪ **트윗 목록을 데이터 프레임으로 변환(twitteR 패키지)**

```
> twListToDF(twList)
```

- **twList** : 트윗 목록

- **list with following components**
 - **# text:** The text of the status
 - **# screenName:** Screen name of the user who posted this status
 - **# id:** ID of this status
 - **# replyToSN:** Screen name of the user this is in reply to
 - **# replyToUID:** ID of the user this was in reply to
 - **# statusSource:** Source user agent for this tweet
 - **# created:** When this status was created

- **# truncated:** Whether this status was truncated
- **# favorited:** Whether this status has been favorited
- **# retweeted:** TRUE if this status has been retweeted
- **# retweetCount:** The number of times this status has been retweeted

```
> # install.packages(c("twitterR", "ROAuth"))
> library(twitterR) # twitter R client
> library(ROAuth)   # for authentication
>
> # twitterR 사용 인증(authentication)
> consumer_key<-"
> consumer_secret<-"
> access_token<-"
> access_secret<-"
```

```
> setup_twitter_oauth(consumer_key,
+   consumer_secret, access_token, access_secret)
[1] "Using direct authentication"
> keyword<-enc2utf8("빅데이터")
> tout<-searchTwitter(keyword, n=1000) # list
> toutDF<-twListToDF(tout)
> names(toutDF)
[1] "text"          "favorited"     "favoriteCount"
[4] "replyToSN"     "created"       "truncated"
[7] "replyToSID"    "id"            "replyToUID"
[10] "statusSource"  "screenName"    "retweetCount"
[13] "isRetweet"     "retweeted"     "longitude"
[16] "latitude"
```



```
> head(toutDF$text)
[1] "RT @aDorable_DN: 180526 그린콘서트
BOOMERANG\\n사실 오늘부터 rest 하려고 했는데,,
넬친 앞에 무릎 꿇고 울면서 포토샵을 컷습니다π
넬친 충성충성\\n#강다니엘 #KangDaniel\\n\\n출
처 : 빅데이터뉴스\\nhttps://t..."
[2] "RT @aDorable_DN: 180526 그린콘서트
BOOMERANG\\n인생은, 강다니엘.\\n#강다니엘
#KangDaniel\\n\\n출처 : 빅데이터뉴스
\\nhttps://t.co/xRxhYbr83R\\n\\nhttps://t.co/Lwmzi
8J0wh\\nhttps://..."
:
```

```
> library(stringr)
> library(KoNLP)
> library(wordcloud)
> useSejongDic()
>
> # 문자 정제
> l<-grepl("가수|브랜드|평판|강다니엘",
+         toutDF$text)
> text<-toutDF$text[!l]
> clr.wd<-gsub("[^A-Za-z가-힣
+ [:space:][:digit:][:punct:]]", "", text)
> clr.wd<-gsub("@|\\n|RT", "", clr.wd)
> clr.wd<-gsub("[[:punct:]]", " ", clr.wd)
> clr.wd<-gsub("[[:digit:]]", "", clr.wd)
```

```

> clr.wd<-tolower(clr.wd)
> clr.wd<-gsub("[a-z]", "", clr.wd)
> clr.wd<-gsub("댓글|https|co|녀모두|11위", "",
clr.wd)
> clr.wd<-gsub("출처|하였습니다|지난달|위에서|
+ 기사", "", clr.wd)
> clr.wd<-str_trim(clr.wd)
>
> lword<-lapply(clr.wd, extractNoun)
> lword<-lapply(lword, function(x) x[nchar(x)>1])
> lword<-do.call(c, lword)
> wd.tbl<-table(lword)
>

```

```

> # 워드 클라우드
> pal<-brewer.pal(8,"Dark2")
> windows()
> wordcloud(names(wd.tbl), wd.tbl, min.freq=3,
+          random.order=F, random.color=T,
+          colors=pal)
> detach("package:wordcloud", unload=TRUE)
> detach("package:KoNLP", unload=TRUE)
> detach("package:stringr", unload=TRUE)
> detach("package:twitter", unload=TRUE)
> detach("package:ROAuth", unload=TRUE)

```

