Pendahuluan

Hal pertama yang perlu dilakukan sebelum menggunakan data yaitu menyiapkan data. Data yang akan kita gunakan biasanya sudah tersimpan dalam komputer. Namun demikian, terkadang data yang akan kita gunakan berasal dari luar komputer dan jenisnya juga bervariasi. Dengan demikian diperlukan kemampuan untuk mengambil data dari berbagai sumber.

Teknik scrabing web merupakan salah satu cara untuk mendapatkan data dari internet. Untuk menggunakan teknik ini, R telah menyiapkan package dengan nama *rvest*. Package tersebut biasa digunakan untuk data extracting, transforming, dan loading. Pelajaran kesatu ini membahasa bagaimana menggunakan packages R untuk membaca data teks dan melakukan scan file. Setelah itu, kita belajar membaca data terstruktur dari database komputer kita dan data excel. Terakhir kita belajar mengambil data dari internet dengan teknik scrape.

1. Mengunduh Open Data

Sebelum melakukan analisis data, langkah pertama yang dilakukan yaitu mengumpulkan data. Salah satu sumber data yang sering digunakan berasal dari open data. Sebagian besar open data dipublish secara terbuka dalam bentuk teks ataupun dalam format API (*Application Programming Interface*). Beberapa website yang menyediakan open data yaitu UCI, Yahoo, Data Football, New York Open Data, dan lain-lain.

Misal kita akan mendownload data pertandingan Liga Pemier Inggris 2017/2018 dari Data Football. Langkah-langkahnya adalah:

a. Buka: http://www.football-data.co.uk/englandm.php
b. Pilih: Season 2017/2018 → Premier League, maka data akan terdownload.

Selain dengan memilih menu Download Data, kita juga bisa langsung download data tersebut langsung dari R Studio. Caranya yaitu dengan menggunakan fungsi: *download.file.* Langkahnya sebagai berikut:

a. Buatlah variabel (misal kita beri nama: *file_bola*) dengan isi link file yang akan didownload.

```
file_bola= "http://www.football-data.co.uk/mmz4281/1718/E0.csv"
```

b. Lakukan download file dengan memasukan variabel yang telah dibuat dan simpan dengan nama: *bola.scv*

```
download.file(file_bola,destfile = "./bola.csv")
```

Selain menggunakan download.file, kita juga bisa menggunakan fungsi *getURL* dari package *RCurl*

a. Install dan load package Rcurl

```
install.packages("RCurl")
library(RCurl)
```

b. Download file menggunakan fungsi: *getURL,* dan simpan dengan nama: *data_bola2*

```
data_bola2 = getURL(file_bola)
```

2. Membaca File

Setelah data pertandingan Liga Pemier Inggris 2017/2018 kita download, selanjutnya yaitu kita akan membaca file tersebut dalam R untuk berlatih.

a. Cek lokasi penyimpanan file yang telah didownload menggunakan *getwd()*

```
getwd()
```

b. Gunakan *read.table* untuk membaca data. Misal kita buat variabel dengan nama: *data_bola*

```
data_bola = read.table("bola.csv",sep = ",", header = TRUE)
```

c. Buat variabel baru dengan nama: *data_bolafilter* dengan memilih kolom: *Date, HomeTeam, AwayTeam, FTHG, FTAG, FTR*

```
data_bolafilter = data_bola[,c("Date","HomeTeam","AwayTeam","FTHG","FTAG","FTR")]
```

d. Apabila file yang kita download sudah dalam format .CSV, maka bisa menggunakan read.csv

```
data_bola = read.csv("bola.csv", header = TRUE)
```

e. Apabila file yang didownload dalam format text seperti *data_bola2*, maka harus diubah terlebih dahulu dalam bentuk tabel.

```
data_bola2_ubah = read.csv(text = data_bola2)
```

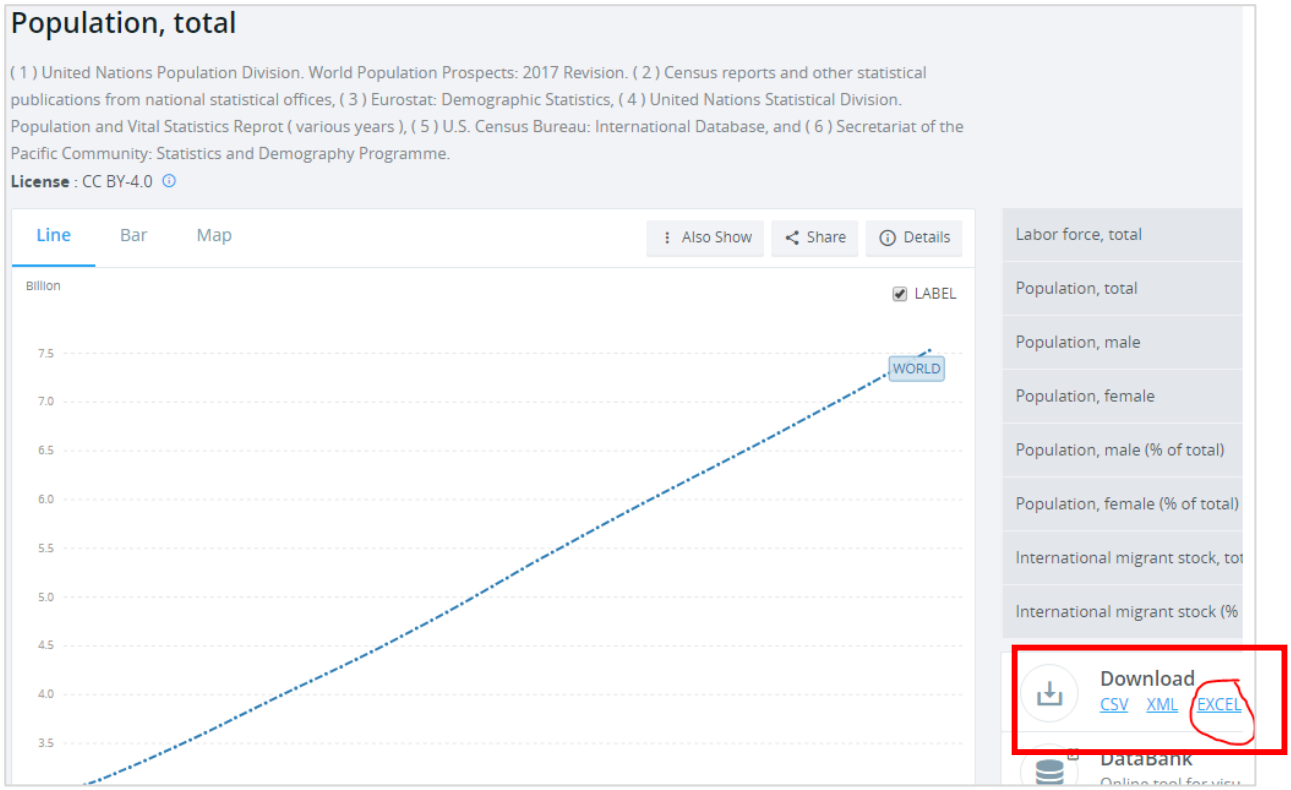
3. Membaca File Excel

File yang kita download terkadang dalam format teks menggunakan Excel. Package *xlsx* digunakan untuk membaca dan memproses file Excel di R.

a. Install dan load package xlxs (pastikan java yang terinstall sesuai dengan type OS, 32/64 bit)

```
install.packages("xlsx")
library(xlsx)
```

b. Download file populasi penduduk dunia dengan mengakses: https://data.worldbank.org/indicator/SP.POP.TOTL kemudian pilih file excel.



c. Setelah didownload, buka file tersebut. Perhatikan sheet-nya dan startrow-nya

	A	B	C	D	E	F	G	H	I
1	Data Source	World Development Indicators							
2	Last Updated Date	1/30/2019							
3									
4	Country Name	Country Code	Indicator Name	Indicator Code	1960	1961	1962	1963	1964
5	Aruba	ABW	Population, tota	SP.POP.TOTL	54211	55438	56225	56695	57
6	Afghanistan	AFG	Population, tota	SP.POP.TOTL	8996351	9166764	9345868	9533954	9731
7	Angola	AGO	Population, tota	SP.POP.TOTL	5643182	5753024	5866061	5980417	6093
8	Albania	ALB	Population, tota	SP.POP.TOTL	1608800	1659800	1711319	1762621	1814
9	Andorra	AND	Population, tota	SP.POP.TOTL	13411	14375	15370	16412	17
10	Arab World	ARB	Population, tota	SP.POP.TOTL	92490932	95044497	97682294	100411076	103239
11	United Arab Emirates	ARE	Population, tota	SP.POP.TOTL	92634	101078	112472	125566	138
12	Argentina	ARG	Population, tota	SP.POP.TOTL	20619075	20953077	21287682	21621840	21953
13	Armenia	ARM	Population, tota	SP.POP.TOTL	1874120	1941491	2009526	2077575	2144
14	American Samoa	ASM	Population, tota	SP.POP.TOTL	20013	20486	21117	21882	22
15	Antigua and Barbuda	ATG	Population, tota	SP.POP.TOTL	55339	56144	57144	58294	59
16	Australia	AUS	Population, tota	SP.POP.TOTL	10276477	10483000	10742000	10950000	11167
17	Austria	AUT	Population, tota	SP.POP.TOTL	7047539	7086299	7129864	7175811	7223
18	Azerbaijan	AZE	Population, tota	SP.POP.TOTL	3895396	4030320	4171425	4315128	4456
19	Burundi	BDI	Population, tota	SP.POP.TOTL	2786106	2839666	2893669	2949926	3010
20	Belgium	BEL	Population, tota	SP.POP.TOTL	9153489	9183948	9220578	9289770	9378
21	Benin	BEN	Population, tota	SP.POP.TOTL	2431622	2465867	2502896	2542859	2585
22	Burkina Faso	BFA	Population, tota	SP.POP.TOTL	4829288	4894580	4960326	5027821	5098
23	Bangladesh	BGD	Population, tota	SP.POP.TOTL	48199747	49592802	51030137	52532417	54129

Data | Metadata - Countries | Metadata - Indicators

d. Baca file tersebut dengan membuat variabel:*populasi*

```
populasi = read.xlsx("populasi_penduduk.xls",sheetIndex=1,startRow=4)
```

e. Buatlah variabel baru *populasi2* yang diambil dari data *populasi* kolom Nama negara, kode negara, dan tahun 2017

```
populasi2 = populasi[,c("Country.Name","Country.Code","X2017")]
```

f. Simpanlah data *populasi2* dalam format excel dengan nama *populasi2017*

g. Cek apakah sudah tersimpan atau belum dengan membuka directory penyimpanan.

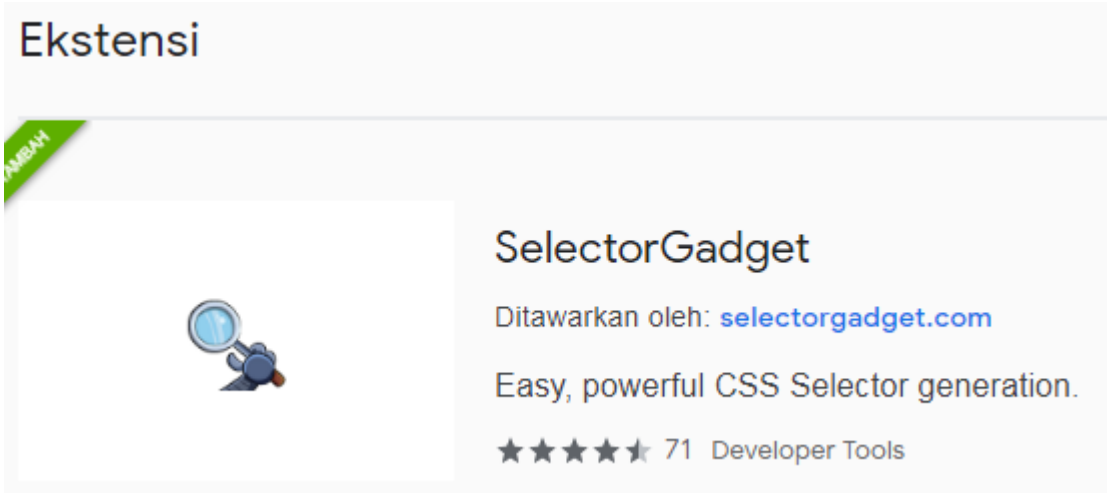
4. Scraping Data Website

Scraping data merupakan teknik yang sering digunakan dalam mengambil data dari suatu website. Kita akan mencoba menggunakan teknik scraping untuk mengambil data suatu portal berita.

a. Pertama, Install dan load package:*rvest*

```
install.packages("rvest")
library(rvest)
```

b. Pasang Selector Gadget pada browser



c. Misal kita scrape halaman: https://news.detik.com/indeks/all?

```
url_detik = "https://news.detik.com/indeks/all?"
```

d. Baca kode HTML-nya

```
page_detik = read_html(url_detik)
```

e. Ambil tanggal postingan dan isi judul berita

```
tanggal_berita = html_nodes(page_detik,".mb5")
Tanggal = html_text(tanggal_berita)
judul_berita = html_nodes(page_detik,"h2")
Judul = html_text(judul_berita)

data.frame(Tanggal,Judul)
list_berita = cbind(Tanggal,Judul)
```

5. Mengakses Data Facebook

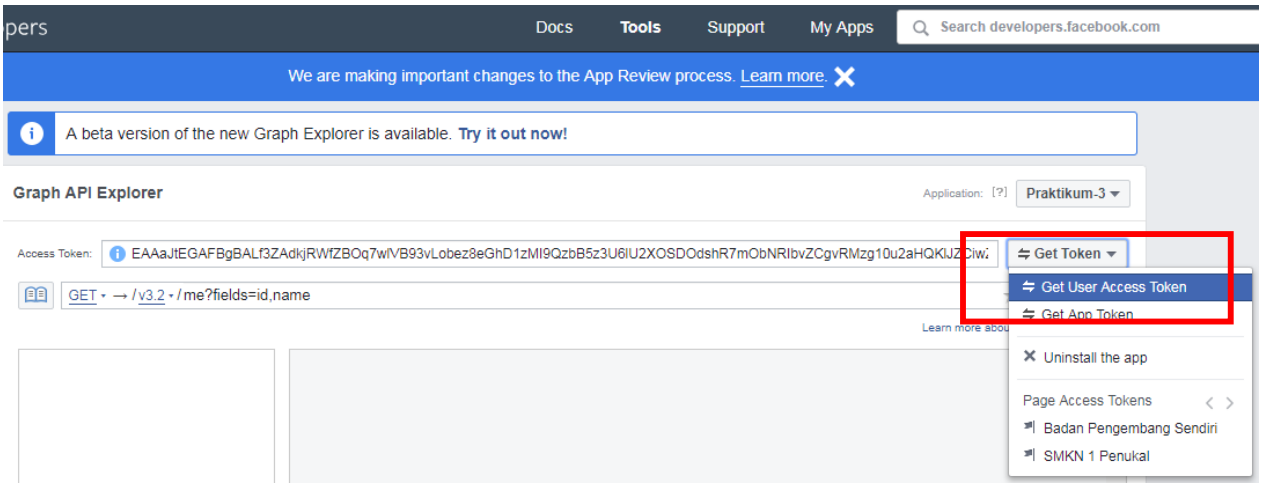
Mendapatkan data di internet selain diperoleh dengan mengakses web open data dan tenik scraping, data juga bisa diperoleh dengan teknik crawling. Teknik crawling memerlukan inspect elemen untuk memperoleh data sedangkan crawling memerlukan API. Untuk mengakses data di Facebook, siapkan package: *Rfacebook.*

a. Install dan load package: *Rfacebook*

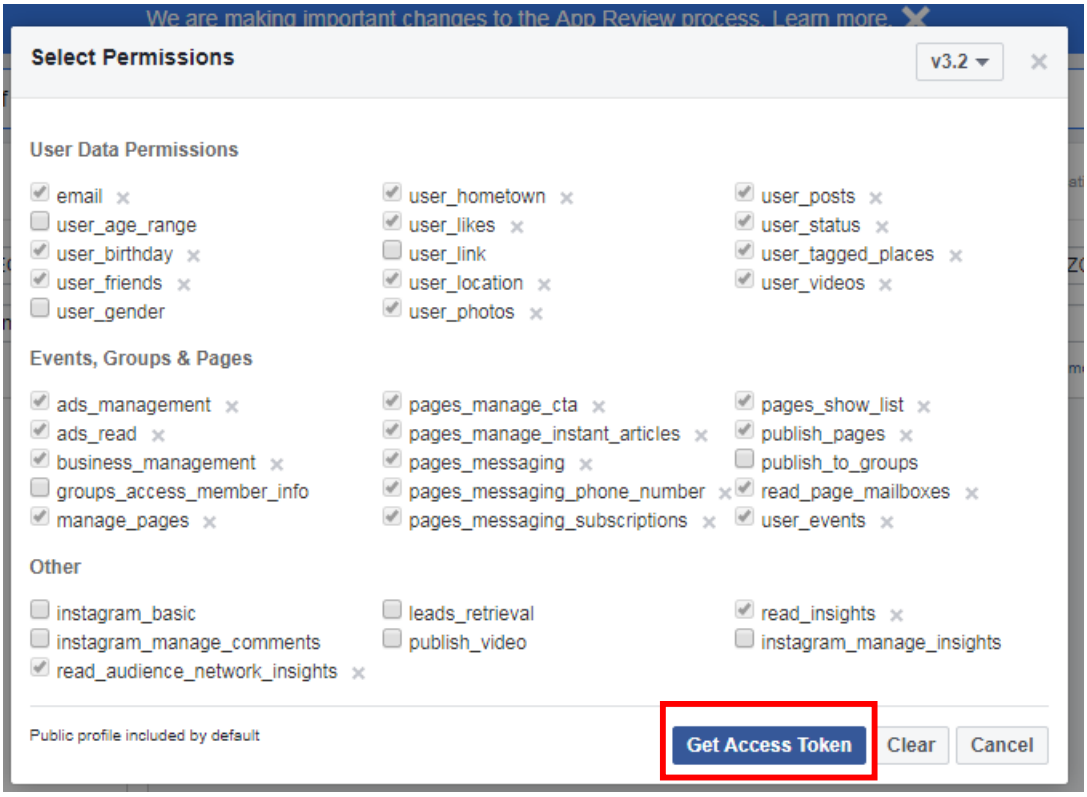
```
install.packages("Rfacebook")
library(Rfacebook)
```

b. Dapatkan token Facebook

- Login Facebook
- Akses halaman: https://developers.facebook.com/tools/explorer/
- Klik Get Token → Get User Access Token



- Klik Get Access Token



- Copy Token

c. Paste dan run Token dalam R. Token sudah bisa digunakan.

```
token = "EAAaJtEGAFBgBAH22GcgMgZAzcwLd2yAE0ZCpCCTSFkz4qD
```

d. Sebagai contoh kita akan mencari page dengan keyword “tahu aci”

```
tahu_aci=searchPages("tahu aci", token, n=100)
```