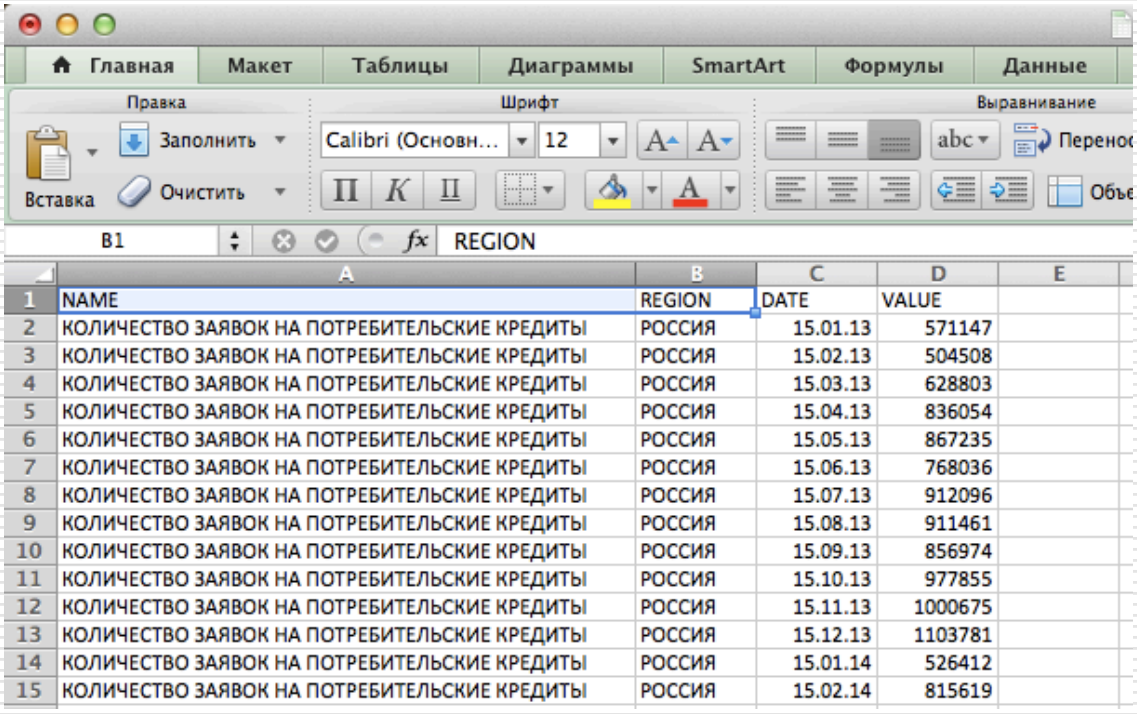
Извлечение полностью структурированных (статических) данных

Копируем данные из http://www.sberbank.com/ru/analytics/opendata



	A	B	C	D	E
1	NAME	REGION	DATE	VALUE	
2	КОЛИЧЕСТВО ЗАЯВОК НА ПОТРЕБИТЕЛЬСКИЕ КРЕДИТЫ	РОССИЯ	15.01.13	571147	
3	КОЛИЧЕСТВО ЗАЯВОК НА ПОТРЕБИТЕЛЬСКИЕ КРЕДИТЫ	РОССИЯ	15.02.13	504508	
4	КОЛИЧЕСТВО ЗАЯВОК НА ПОТРЕБИТЕЛЬСКИЕ КРЕДИТЫ	РОССИЯ	15.03.13	628803	
5	КОЛИЧЕСТВО ЗАЯВОК НА ПОТРЕБИТЕЛЬСКИЕ КРЕДИТЫ	РОССИЯ	15.04.13	836054	
6	КОЛИЧЕСТВО ЗАЯВОК НА ПОТРЕБИТЕЛЬСКИЕ КРЕДИТЫ	РОССИЯ	15.05.13	867235	
7	КОЛИЧЕСТВО ЗАЯВОК НА ПОТРЕБИТЕЛЬСКИЕ КРЕДИТЫ	РОССИЯ	15.06.13	768036	
8	КОЛИЧЕСТВО ЗАЯВОК НА ПОТРЕБИТЕЛЬСКИЕ КРЕДИТЫ	РОССИЯ	15.07.13	912096	
9	КОЛИЧЕСТВО ЗАЯВОК НА ПОТРЕБИТЕЛЬСКИЕ КРЕДИТЫ	РОССИЯ	15.08.13	911461	
10	КОЛИЧЕСТВО ЗАЯВОК НА ПОТРЕБИТЕЛЬСКИЕ КРЕДИТЫ	РОССИЯ	15.09.13	856974	
11	КОЛИЧЕСТВО ЗАЯВОК НА ПОТРЕБИТЕЛЬСКИЕ КРЕДИТЫ	РОССИЯ	15.10.13	977855	
12	КОЛИЧЕСТВО ЗАЯВОК НА ПОТРЕБИТЕЛЬСКИЕ КРЕДИТЫ	РОССИЯ	15.11.13	1000675	
13	КОЛИЧЕСТВО ЗАЯВОК НА ПОТРЕБИТЕЛЬСКИЕ КРЕДИТЫ	РОССИЯ	15.12.13	1103781	
14	КОЛИЧЕСТВО ЗАЯВОК НА ПОТРЕБИТЕЛЬСКИЕ КРЕДИТЫ	РОССИЯ	15.01.14	526412	
15	КОЛИЧЕСТВО ЗАЯВОК НА ПОТРЕБИТЕЛЬСКИЕ КРЕДИТЫ	РОССИЯ	15.02.14	815619	

Извлечение полностью структурированных (статических) данных

```
1  library("readxl")
2  sd <- read_excel("/Users/mac/Desktop/LABS/sbData.xlsx")
3  wh01 <- unique(sd$NAME) # параметры (15)
4  #[1] "КОЛИЧЕСТВО ЗАЯВОК НА ПОТРЕБИТЕЛЬСКИЕ КРЕДИТЫ"   "СРЕДНЯЯ СУММА ЗАЯВКИ НА ПОТРЕБИТЕЛЬСКИЙ КРЕДИТ"
5  #[3] "КОЛИЧЕСТВО ЗАЯВОК НА ИПОТЕЧНЫЕ КРЕДИТЫ"        "СРЕДНЯЯ СУММА ЗАЯВКИ НА ИПОТЕЧНЫЙ КРЕДИТ"
6  #[5] "КОЛИЧЕСТВО НОВЫХ ДЕПОЗИТОВ"                    "СРЕДНЯЯ СУММА НОВОГО ДЕПОЗИТА"
7  #[7] "СРЕДНЯЯ ЗАРПЛАТА"                              "СРЕДНЯЯ ПЕНСИЯ"
8  #[9] "В СРЕДНЕМ РУБ. НА ТЕКУЩЕМ СЧЕТЕ НА ЧЕЛОВЕКА"   "В СРЕДНЕМ ДЕПОЗИТОВ В РУБ. НА ЧЕЛОВЕКА"
9  #[11] "СРЕДНИЕ РАСХОДЫ ПО КАРТАМ"                    "СРЕДНИЙ ЧЕК В ФОРМАТЕ ФАСТФУД"
10 #[13] "СРЕДНИЙ ЧЕК В ФОРМАТЕ РЕСТОРАН"               "СРЕДНИЕ ТРАТЫ В РЕСТОРАНЕ ФАСТФУД"
11 #[15] "СРЕДНИЕ ТРАТЫ В РЕСТОРАНЕ"
12 wh02 <- unique(sd["REGION"]) # регионы (84)
```

Извлечение полностью структурированных (статических) данных

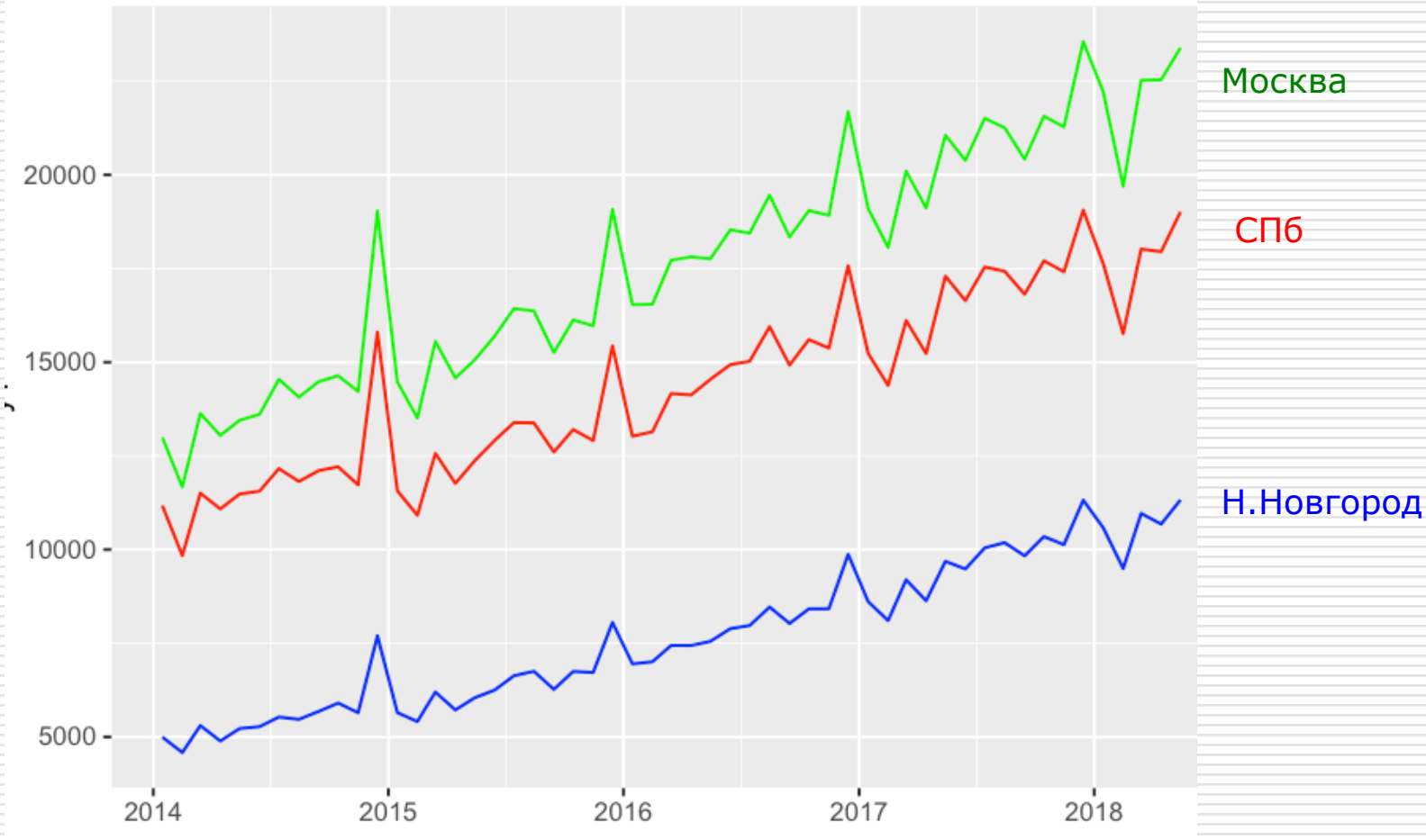
```
sd <- read_excel("/Users/mac/Desktop/LABS/sbData.xlsx")
wh01 <- unique(sd$NAME) # параметры (15)
wh02 <- unique(sd["REGION"]) # регионы (84)
wh03 <- unique(sd[3]) # даты (65)

spb1 <- sd$REGION[sd$REGION == "САНКТ-ПЕТЕРБУРГ"]
y1 <- sd$VALUE[sd$REGION == "САНКТ-ПЕТЕРБУРГ" & sd$NAME == "СРЕДНИЕ РАСХОДЫ ПО КАРТАМ"]
y2 <- sd$VALUE[sd$REGION == "МОСКВА" & sd$NAME == "СРЕДНИЕ РАСХОДЫ ПО КАРТАМ"]
x <- sd$DATE[sd$REGION == "САНКТ-ПЕТЕРБУРГ" & sd$NAME == "СРЕДНИЕ РАСХОДЫ ПО КАРТАМ"]
df <- data.frame(x,y1,y2)

y3 <-sd$VALUE[sd$REGION == "НИЖЕГОРОДСКАЯ ОБЛАСТЬ" & sd$NAME == "СРЕДНИЕ РАСХОДЫ ПО КАРТАМ"]

# https://r-datascience.ru/ggplot2_guide/
library("ggplot2")
ggplot(df, aes(x)) +
  geom_line(aes(y=y1), colour="red") +
  geom_line(aes(y=y2), colour="green") +
  geom_line(aes(y=y3), colour="blue")
```

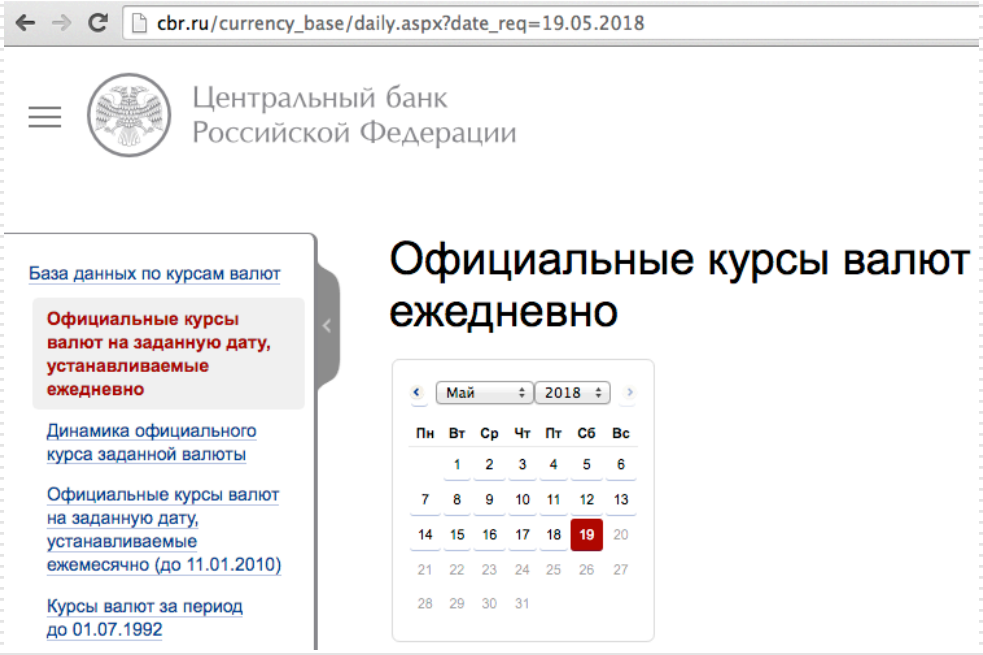
Средние расходы по картам Сбербанка



Извлечение полностью структурированных (динамических) данных

http://cbr.ru/currency_base/daily.aspx

```
library("rvest")
library("ggplot2”)
```



Параметры функции:

Rates <- function(BaseURL, BegDate, EndDate, Curr)

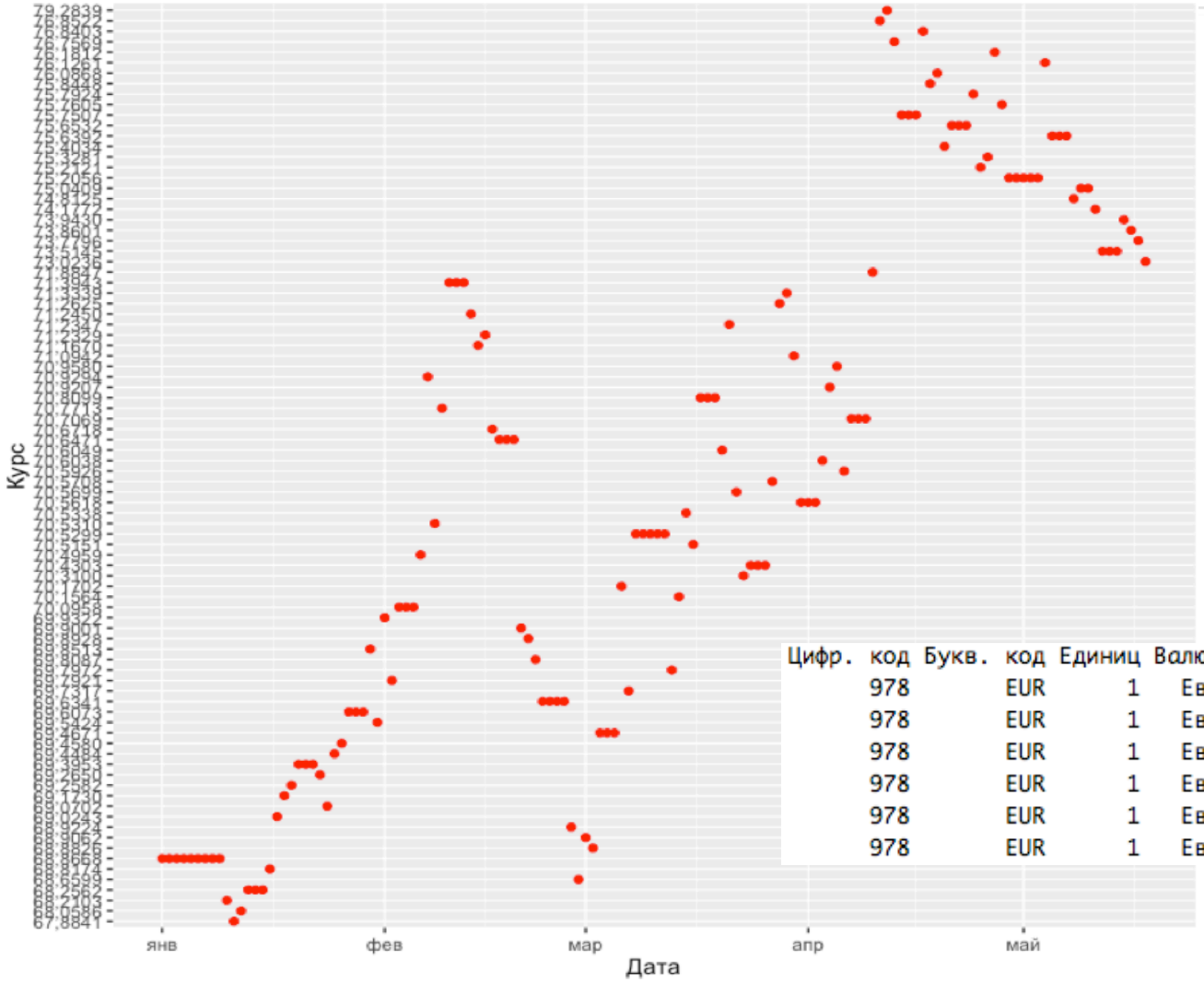
```
library("rvest")
library("ggplot2")

Rates <- function(BaseURL, BegDate, EndDate, Curr) {
  bd <- as.Date(BegDate)
  ed <- as.Date(EndDate)
  vDate <- seq.Date(bd, ed, 1)
  Currency <- toupper(Curr)
  fCourse = NULL

  len <- length(vDate)
  for (dd in 1:len)
  {
    locURL <- paste0(BaseURL, vDate[dd])
    docSource <- read_html(locURL)
    table <- html_table(docSource)
    tab <- table[[1]]
    fCourse <- rbind(fCourse,cbind(subset(tab, tab[2]==Currency),
dDate=vDate[dd]))
    Sys.sleep(3)
  }
  fCourse
}
```

```
z <- Rates("http://cbr.ru/currency_base/daily.aspx?date_req=", "2018-01-01", "2018-05-18", "eur")

ggplot(data=z, aes(x=z$"dDate",y=z$"Курс")) + geom_point(color = "red") + labs (x="Дата",y="Курс")
```



Цифр. код	Букв. код	Единиц	Валюта	Курс	dDate
978	EUR	1	Евро	72,5806	2018-06-01
978	EUR	1	Евро	72,7183	2018-06-02
978	EUR	1	Евро	72,7183	2018-06-03
978	EUR	1	Евро	72,7183	2018-06-04
978	EUR	1	Евро	72,4384	2018-06-05
978	EUR	1	Евро	72,5130	2018-06-06

Пакет *Rcrawler* (основная функция)

Rcrawler*(Website, no_cores , nbcon, MaxDepth, DIR, RequestsDelay = 0, duplicatedetect = FALSE, Obeyrobots = FALSE, IndexErrPages , Useragent , Timeout = 5, URLlenlimit = 255, urlExtfilter , urlregexfilter , ignoreUrlParams, statslinks = FALSE, Encod, patterns, excludepattern)*

Website
no_cores
nbcon
MaxDepth
DIR
RequestDelay = 0
duplicatedetect = FALSE
Obeyrobots = FALSE
IndexErrPages

UserAgent
Timeout = 5
URLlenlimit = 255
urlExtfilter
ignoreUrlParams
statslinks = FALSE
Encod
Patterns
excludepattern

Пакет *Rcrawler* (основная функция)

- *Rcrawler("http: / /www.example.com/")*

Анализирует, индексирует и запоминает страницы используя конфигурацию по умолчанию.

- *Rcrawler(Website = "http:/ /www.example.com/" , no_cores = 8, nbcon=8, Obeyrobots = TRUE, Useragent="Mozilla 3.11")*

Анализирует и индексирует сайт используя 8 ядер и 8 параллельных запросов согласуясь с правилами *robot.txt*.

- *Rcrawler(Website = "http:/ /www.example.com/" , no_cores = 4, nbcon = 4, urlregexfilter = "/\\d{4}/\\d{2}/", DIR = " . /myrepo" , MaxDepth=3)*

Анализирует и индексирует сайт используя 4 ядра и 4 параллельных запроса. Однако индексирует только URLs удовлетворяющие регулярному выражению (/\\d{4}/\\d{2}/), и запоминает страницы в пользовательском директории ''myrepo''. Анализ заканчивается по достижении 3 уровня.

- *Rcrawler(Website = "http:/ /www.example.com/" , urlregexfilter = "/ \\d{4}/ \\d{2}/" , patterns = c("//∗[@class='post−body entry−content']","//∗[@class ='post−title entry−title ']"))*

Анализирует и индексирует только URLs удовлетворяющие регулярному выражению (/\\d{4}/\\d{2}/) и выскабливает контент удовлетворяющий двум XPath/.

Пакет *Rcrawler* (вспомогательные функции)

Takes a URL as input, fetches its web page, and extracts all links following a set of rules.
LinkExtractor*(url , id , lev , IndexErrPages , Useragent , Timeout = 5, URLlenlimit = 255, urlExtfilter , statslinks = FALSE, encod, urlbotfiler , removeparams)*

Transforms a list of URLs into a canonical form
LinkNormalization *(links , current)*

Parses a web page and retrieves the character en- coding based on the content and HTTP header.
Getencoding *(url)*

Fetches and parses robots.txt file and returns its corresponding access rules.
RobotParser*(website, useragent)*

Generates SimHash fingerprint of a given web page, using an external Java class.
getsimHash*(string , hashbits)*

Extracts URL parameters and values from a given URL.
Linkparameters *(URL)*

Excludes a given set of parameters from a specific URL.
Linkparamsfilter *(URL, params)*

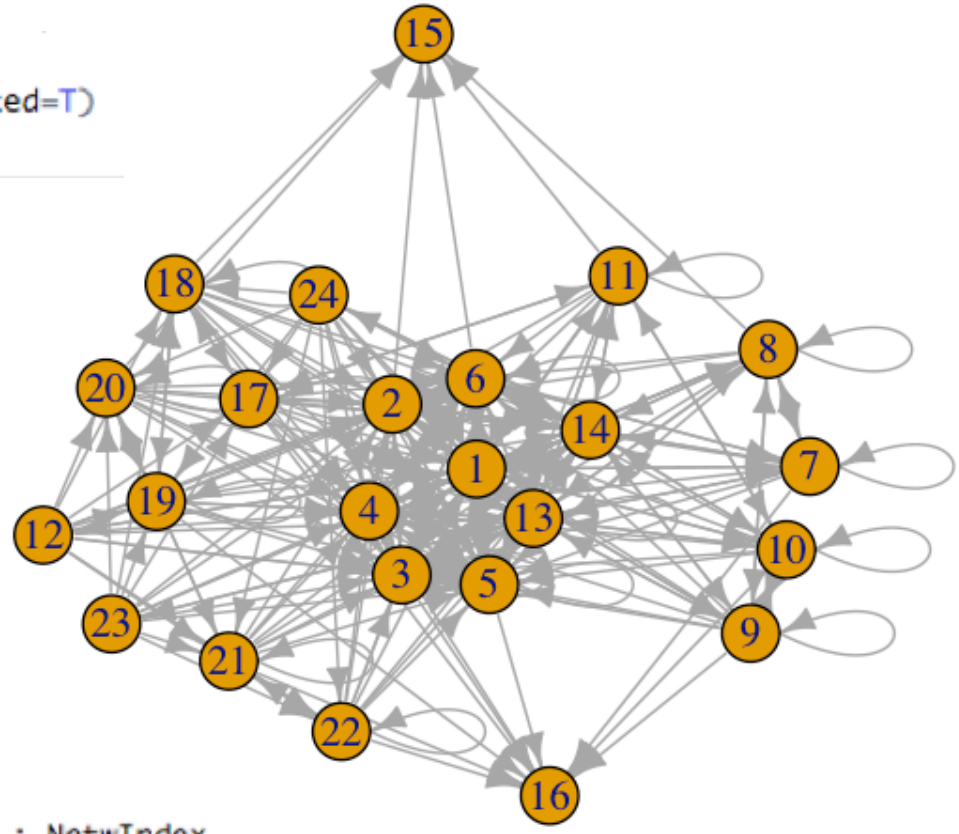
Extracts contents matching a given set of XPath patterns.
contentscraper *(webpage , patterns , patnames , excludepat , astext = TRUE, encod)*

Пакет *Rcrawler* (основная функция)

```
> Rcrawler(Website = "http://glofile.com/" , NetworkData = TRUE)
In process : 1..
Progress: 7.69 %  :  1  parssed from  13  | Collected pages: 1  | Level: 1
In process : 2..3..4..
Progress: 12.50 %  :  2  parssed from  16  | Collected pages: 4  | Level: 1
In process : 5..6..7..
Progress: 31.25 %  :  5  parssed from  16  | Collected pages: 7  | Level: 1
In process : 8..9..10..
Progress: 50.00 %  :  8  parssed from  16  | Collected pages: 10  | Level: 1
In process : 11..12..13..
Progress: 47.83 %  :  11  parssed from  23  | Collected pages: 13  | Level: 2
In process : 14..15..16..
Progress: 58.33 %  :  14  parssed from  24  | Collected pages: 14  | Level: 2
In process : 17..18..19..
Progress: 70.83 %  :  17  parssed from  24  | Collected pages: 17  | Level: 2
In process : 20..21..22..
Progress: 83.33 %  :  20  parssed from  24  | Collected pages: 20  | Level: 2
In process : 23..24..
Progress: 95.83 %  :  23  parssed from  24  | Collected pages: 22  | Level: 3
+ Check INDEX dataframe variable to see crawling details
+ Collected web pages are stored in Project folder
+ Project folder name : glofile.com-311416
+ Project folder path : /Users/mac/Desktop/R_script/glofile.com-311416
+ Network nodes are stored in a variable named : NetwIndex
+ Network eadges are stored in a variable named : NetwEdges
```

Обработка *NetwEdges*

```
library(igraph)
network<-graph.data.frame(NetwEdges, directed=T)
plot(network)
```




```
Network nodes are stored in a variable named : NetwIndex
Network eadges are stored in a variable named : NetwEdges
```

Резюме

- Веб-скрапинг (*Web Scraping*) - совокупность методов получения интересующего контента с небольшими затратами.
- Веб-скрапинг широко используемая технология поиска неструктурированной информации.