# WEB Scraping

## Extraer información a partir de una página WEB

Actualmente el internet nos proporciona una fuente inagotable de información, y el poder acceder a la misma de forma sencilla y rápida es de.vital importancia si se quiere estar actualizado ó al tanto de un tema. La información que se consigue en internet viene dada en páginas web las cuales principalmente están escritas en código HTML, el cual es un lenguaje que reconocen y traducen los navegadores como Google Chrome , safari o Mozilla Firefox.

El web scraping es un proceso mediante el cual es posible extraer información a partir del código HTML de una página web, en principio es posible extraer cual texto que se encuentre en dicha página web. En particular es posible extraer información como el precio del dólar de alguna fuente oficial de información, como lo puede ser el BCV, o también es posible tener fuentes no oficiales como dolar today o Airtm.

### Proceso usando R

A continuación se explicará el proceso de extracción de información a partir de una pagína web usando **R**. Para este fin, se utilizarán los paquetes **rvest** y **xml2**, los cuales cuentan con una serie de funciones que nos permiten extraer el contenido del código **HTML** del cual se forma cualquier página WEB. Dentro de las funciones que se utilizarán se encuentran,

- **read_html**: función perteneciente al paquete **xml2**, la cual permite leer el código HTML de la página deseada, como principal argumento usa la dirección URL de la página en cuestión.
- **html_nodes**: función perteneciente al paquete **rvest**, la misma permite la extracción de partes específicas del código HTML, usando un selector de CSS.
- **xml_child**: función perteneciente al paquete **xml2**, la misma permite extraer un elemento en específico del código analizado, como principales argumentos se encuentran,
  - x : conjunto de nodos disponibles en la página WEB.
  - search : número de nodo al cual se quiere acceder.
- **xml_attrs**: función perteneciente al paquete **rvest**, la cual permite extraer y acceder a los atributos del código en cuestión, como principal argumento usa un conjunto de nodos, el cual se genera a partir de la función **read_html**.

Mediante el uso de la funciones anterioremente explicadas es posible extraer cualquier información que esté contenida en una página WEB, en particular en este caso nos centraremos en extraer el precio del Dolar de tres diferentes páginas WEB, las cuales son,

1) **Página Oficial del Banco central de Venezuela**: en esta página se puede consultar el precio del dolar oficial en Venezuela según el BCV. Para consultar el precio presione aquí.
2) **Página de tasas de Airtm**: en esta página se puede consultar el precio de compra, promedio y precio de venta del Dolar Airtm, la cual es una plataforma que permite trasformar esta moneda ($) a otras monedas de una gran cantidad de paises. Para consultar el precio presione aquí.
3) **Página de DolarToday**: en esta página se puede consultar el precio de DolarToday. Para consultar el precio presione aquí.

Tanto para la primera como para la tercera página el precio se ubica en la parte derecha de la misma. Por otra parte la página de Airtm muestra la información en el centro al inicio de la página. Es importante señalar que al realizar esta consulta el precio devuelto será el precio actual del dolar, usando este procedimiento es posible elaborar un histórico de estos precios.

## Extraer precio Dolar Dicom

Para este ejemplo se extraerá el precio actual de Dolar Oficial Dicom, el cual se encuentra en la página del BCV. La librería para lograr este fin es "rvest",

```
#cargo librería a usar
library(rvest)
```

```
## Loading required package: xml2
```

Una vez cargada la librería, le indico a la función "read_html" la página Web que se quiere explorar,

```
#ingreso página web a analizar
webpage <- read_html("http://www.bcv.org.ve")
```

Luego de esto, usando la función **html_nodes** es posible extraer del código html de la página los contenedores donde se encuentra la información deseada,

```
#obtengo precio compra
results <- webpage %>% html_nodes("div")
```

Despues de tener el conjunto de nodos definido usamos la función **xml_child** con el fin de extraer información específica de cada nodo. Una vez ubicada la información la extraemos y la guardamos en la variable "b",

```
#selecciono la tasa de Cambio $/BsS
b <- as.character(xml_child(xml_child(xml_child(results[[66]], 7), 1), 1))

substr(b, 1,70)
```

```
## [1] "<div class=\"row recuadrotsmc\">\n\t\t\t<div class=\"col-sm-6 col-xs-6\">\n  \t\t"
```

```
substr(b, 71,140)
```

```
## [1] "\t<img src=\"/sites/default/files/dollar-04_2.png\" class=\"icono_bss_blan"
```

```
substr(b, 141,210)
```

```
## [1] "co\"><span> Bs/USD</span>\t </div>\n                              <div class="
```

```
substr(b, 211,300)
```

```
## [1] "\"col-sm-6 col-xs-6\">\n<strong> 73.193,31 </strong> </div>\n\t        </div>"
```

Como se puede apreciar la información deseada se encuentra en dicha variable, pero con mucha mas información, por tal motivo es necesario extraer sólo el valor numérico de la tasa deseada. Para ello primero ubico la palabra "strong" para así conocer la ubicación de la misma con respecto a la cadena de caracteres que se encuentra en la variable "b",

```
#extraigo información importante
b1 <- regexpr('strong', b)[1]
b1
```

```
## [1] 233
```

Una vez conocida esta posición, usando la función “substr” del paquete base, es posible extraer el precio deseado. Es importante señalar que en este caso los valores “8” y “16” son los espacios necesarios para que el valor de la tasa sea extraida de manera satisfactoria.

```
b2 <- substr(b, b1+8, b1+16)
b2
```

```
## [1] "73.193,31"
```

Luego de estraer la tasa, a la misma se le debe hacer una transformación pues es necesario cambiar la “,” por el “.”, para así obtener un valor numérico que reconozca R. Para ello primero se debe eliminar el “.” que se utiliza como separador de miles, este caracter se reemplaza por el caracter vacio "“. Despues de esto la”," se reemplaza por el “.”.

```
b3 <- gsub("\\.", "",b2)
b3
```

```
## [1] "73193,31"
```

```
b4 <- gsub(",", ".",b3)
b4
```

```
## [1] "73193.31"
```

Finalmente obtengo el valor numérico de la tasa que deseo conocer,

```
b5 <- as.numeric(b4)
b5
```

```
## [1] 73193.31
```

### Extraer precio Dolar Airtm

Para realizar esta extracción el procedimiento es similar al explicado anteriormente, primero se debe pasarle a la función **read_html** la página web a revisar, luego se ubica la información a extraer, finalmente se extrae la información y se convierte en un valor numérico,

```
#Extraigo precio Airtm

#modifico página web donde se realizará la busqueda
webpage <- read_html("https://rates.airtm.io")

#obtengo precio compra
a <- xml_attrs(xml_child(xml_child(webpage, 1), 4))[["content"]]
```

```
#extraigo información importante
a1 <- regexpr('T.C:', a)[1]
a1
```

```
## [1] 18
```

```
a2 <- as.numeric(substr(a, a1+5, a1+12))

#imprimo tasa de compra del $/BsS
a2
```

```
## [1] 70312
```

Es importante señalar que los valores extraidos fueron transformados a número con el fin de realizar operaciones con los mismos. De hecho este proceso se puede repetir cada hora y así crear un histórico a partir del cual se puede generar brechas y gráficos que pueden resultar muy interesantes e informativos.

### Extraer precio Dolar Today

Para realizar esta extracción el procedimiento es similar al explicado anteriormente,

```
#Extraigo precio Dolar Today

#modifico página web donde se realizará la busqueda
webpage <- read_html("https://dolartoday.com/")

#obtengo precio DolarToday
a <- (xml_child(xml_child(webpage, 1), 15))

#extraigo información importante
a1 <- regexpr('Bs.', a)[1]
a1
```

```
## [1] 90
```

```
a2 <- as.numeric(gsub(",", ".",substr(a, a1+4, a1+11)))

#imprimo tasa de compra del $/BsS
a2
```

```
## [1] 75201.82
```

De esta manera se ha logrado extraer información importante relacionada con el precio oficial y no oficial del dólar. Esto se puede realizar con finés meramente informativos, es decir, para saber el precio del dólar en cualquier momento ó también se puede usar para elaborar una base de datos y así poder observar el comportamiento del precio del dólar durante un tiempo determinado. Como próximos pasos se propone usar esta técnica para extraer información de Twitter.

Para mayor comodidad puede revisar mi repositorio en el siguiente enlace.