How to dawnload musics with a site links.

Mehrab Atighi

5/31/2022

Introduction to Web Mining

What is Web Mining?

Web Mining is a process based on data mining techniques in which information is extracted from documents, Internet services. The main purpose of web mining can be to discover useful information from the World Wide Web and its usage patterns.

Introduction to Web Scraping

what is Web Scraping?
Web Scraping is the process of using bots to extract content and data from a website.
In addition to content, web scraping can also extract HTML code elements and publish that information wherever needed.

Html source

each website all over the world have a html source that we can see that. for example we open a site with ‘https://download1music.ir/song-without-words/‘ address. and we want to see this site html sorce.
for recive the html source of sites we should right click on a website and select a View page source or push ctrl + U (in google chorome browser).

Html source

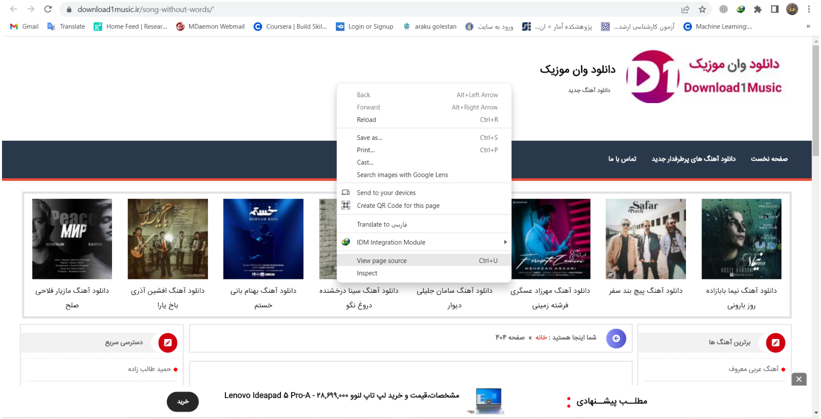
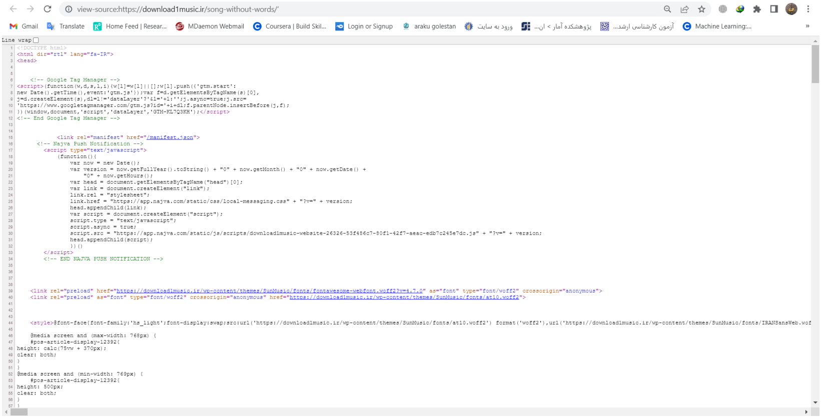


Figure 1: how to open view page source of site with right click

Html source



```
<!DOCTYPE html>
<html dir="rtl" lang="fa-IR">
<head>


    <!-- Google Tag Manager -->
<script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({'gtm.start':
new Date().getTime(),event:'gtm.js'});var f=d.getElementsByTagName(s)[0],
j=d.createElement(s),dl=l!='dataLayer'?'&l='+l:'';j.async=true;j.src=
'https://www.googletagmanager.com/gtm.js?id='+i+dl;f.parentNode.insertBefore(j,f);
})(window,document,'script','dataLayer','GTM-KL7Q3KH');</script>
<!-- End Google Tag Manager -->


            <link rel="manifest" href="/manifest.json">
      <!-- Najva Push Notification -->
        <script type="text/javascript">
            (function(){
                var now = new Date();
                var version = now.getFullYear().toString() + "0" + now.getMonth() + "0" + now.getDate() +
                    "0" + now.getHours();
                var head = document.getElementsByTagName("head")[0];
                var link = document.createElement("link");
                link.rel = "stylesheet";
                link.href = "https://app.najva.com/static/css/local-messaging.css" + "?v=" + version;
                head.appendChild(link);
                var script = document.createElement("script");
                script.type = "text/javascript";
                script.async = true;
                script.src = "https://app.najva.com/static/js/scripts/download1music-website-26326-53f486c7-80f1-42f7-aeac-edb7c245e7dc.js" + "?v=" + version;
                head.appendChild(script);
                })()
        </script>
        <!-- END NAJVA PUSH NOTIFICATION -->




    <link rel="preload" href="https://download1music.ir/wp-content/themes/SunMusic/fonts/fontawesome-webfont.woff2?v=4.7.0" as="font" type="font/woff2" crossorigin="anonymous">
    <link rel="preload" as="font" type="font/woff2" crossorigin="anonymous" href="https://download1music.ir/wp-content/themes/SunMusic/fonts/at10.woff2">



    <style>@font-face{font-family:'hs_light';font-display:swap;src:url('https://download1music.ir/wp-content/themes/SunMusic/fonts/at10.woff2') format('woff2'),url('https://download1music.ir/wp-content/themes/SunMusic/fonts/IRANSansWeb.wof

    @media screen and (max-width: 768px) {
    #pos-article-display-12392{
height: calc(75vw + 370px);
clear: both;
}
}
@media screen and (min-width: 769px) {
    #pos-article-display-12392{
height: 500px;
clear: both;
}
}
```

Figure 2: a source page of site

Whats our purpose?

We want to automatic download all musics of bottom address with using R language. ‘https://download1music.ir/song-without-words/‘ so at the first we should install some packages in R.

Needed Packages:

- rvest
- tidyverse
- stringi

Installing

```
#install.packages("rvest")
#install.packages("tidyverse")
#install.packages("stringi")
```

Start Web Scraping:

we should library needed packages after installing them.

library Packages

```
library(rvest)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter()         masks stats::filter()
## x readr::guess_encoding() masks rvest::guess_encoding()
## x dplyr::lag()            masks stats::lag()
```

```
library(stringi)
```

Web Scraping

at the first we should give URL of site to R.

```
URL <- "https://download1music.ir/song-without-words/"
```

now we should read html source of url with read_html function.

```
pg <- read_html(URL)
```

Web Scraping

Now we want to see pg:

```
pg
```

```
## {html_document}
## <html dir="rtl" lang="fa-IR">
## [1] <head>\n<meta http-equiv="Content-Type" content="text/html; charset=UTF-8 ...
## [2] <body data-rsssl="1">\n<noscript><iframe src="https://www.googletagmanage ...
```

Web Scraping

Now we should find nodes that are starting with a tag in Html source codes. for do this we need to use html_nodes function that the first arg should be a html of a link and the second is tagged that we want to find that.

```
u1 = html_nodes(pg, "a")
head(u1)
```

```
## {xml_nodeset (6)}
## [1] <a class="hvr-buzz" href="https://download1music.ir/" target="_self" titl ...
## [2] <a href="https://download1music.ir/" target="_self" title="<U+062F><U+0627><U+0646><U+0644><U+0648>
## [3] <a href="/"><U+0635><U+0641><U+062D><U+0647> <U+0646><U+062E><U+0633><U+062A></a>
## [4] <a href="https://download1music.ir/category/top-songs/"><U+062F><U+0627><U+0646><U+0644><U+0648><U-
## [5] <a href="https://download1music.ir/%d8%aa%d9%85%d8%a7%d8%b3-%d8%a8%d8%a7- ...
## [6] <a href="https://download1music.ir/%d9%85%d9%87%d8%af%db%8c-%d8%ac%d9%85% ...
```

Web Scraping

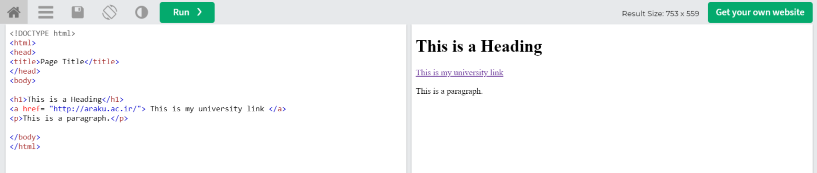
Now we want to find a files that refrenced by "href" taged in a tag of html source. for do it we need to use html_attr function. that the first arg is our nodes and the second arg is sth that we need to find that.

```
u2 = html_attr(html_nodes(pg, "a"), "href")
#u2 = html_attr(u1, "href")
head(u2)
```

```
## [1] "https://download1music.ir/"
## [2] "https://download1music.ir/"
## [3] "/"
## [4] "https://download1music.ir/category/top-songs/"
## [5] "https://download1music.ir/%d8%aa%d9%85%d8%a7%d8%b3-%d8%a8%d8%a7-%d9%85%d8%a7/"
## [6] "https://download1music.ir/%d9%85%d9%87%d8%af%db%8c-%d8%ac%d9%85%d8%a7%d9%84%db%8c-%d9%87%d9%88%d8%
```

Whats a Tag and href in HTML?

if you pay attention to bottom picture you can understand whats a tag and href?
its a tag that will hyper refrence youre text to a link that if you click on that new webpage will be open for you.



```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>This is a Heading</h1>
<a href= "http://araku.ac.ir/"> This is my university link </a>
<p>This is a paragraph.</p>

</body>
</html>
```

This is a Heading

This is my university link

This is a paragraph.

Figure 3: html codes that include a tag with href.

Web Scraping

So now we can make a tibble :

```
links <- tibble(url=html_attr(html_nodes(pg, "a"), "href"))
head(links)
```

```
## # A tibble: 6 x 1
##   url
##   <chr>
## 1 https://download1music.ir/
## 2 https://download1music.ir/
## 3 /
## 4 https://download1music.ir/category/top-songs/
## 5 https://download1music.ir/%d8%aa%d9%85%d8%a7%d8%b3-%d8%a8%d8%a7-%d9%85%d8%a7/
## 6 https://download1music.ir/%d9%85%d9%87%d8%af%db%8c-%d8%ac%d9%85%d8%a7%d9%84%d~
```

Web Scraping

Now we want to filter our data (links tibble) that at the 4 string of url is equal to ".mp3" or its as a mp3 file. the str_sub function is selecting each url and the distinct function will be uniqe our data (with out any repeat!).

```
links <- links %>%
  filter(str_sub(url, -4) == ".mp3") %>%
  distinct()
print(links)
```

```
## # A tibble: 53 x 1
##    url
##    <chr>
##  1 https://dl.download1music.ir/Music/Without-Words/0bikalam download1music.ir ~
##  2 https://dl.download1music.ir/Music/Without-Words/0bikalam download1music.ir ~
##  3 https://dl.download1music.ir/Music/Without-Words/0bikalam download1music.ir ~
##  4 https://dl.download1music.ir/Music/Without-Words/0bikalam download1music.ir ~
##  5 https://dl.download1music.ir/Music/Without-Words/0bikalam download1music.ir ~
##  6 https://dl.download1music.ir/Music/Without-Words/0bikalam download1music.ir ~
##  7 https://dl.download1music.ir/Music/Without-Words/0bikalam download1music.ir ~
##  8 https://dl.download1music.ir/Music/Without-Words/0bikalam download1music.ir ~
##  9 https://dl.download1music.ir/Music/Without-Words/0bikalam download1music.ir ~
## 10 https://dl.download1music.ir/Music/Without-Words/0bikalam download1music.ir ~
## # ... with 43 more rows
```

Web Scraping

Now we want to select the file name from url (first column of links tibble).

for do this we should use stri_sub function that the first arg is data and the second arg is a index that we start from that to end of string.

the stri_locate_last function is selecting the last `Slash` of url and +1 is the next character after last `Slash`.

so here we are making a new column as file_name in links tibble that shows the name of each file.

```r
links$file_name <- stri_sub(links$url, from = (stri_locate_last(links$url, fixed="/")[,1]+1))
```

Web Scraping

Now we want to see the links tibble columns here:

```
links[1:6 , 1]
```

```
## # A tibble: 6 x 1
##   url
##   <chr>
## 1 https://dl.download1music.ir/Music/Without-Words/Obikalam download1music.ir (~
## 2 https://dl.download1music.ir/Music/Without-Words/Obikalam download1music.ir (~
## 3 https://dl.download1music.ir/Music/Without-Words/Obikalam download1music.ir (~
## 4 https://dl.download1music.ir/Music/Without-Words/Obikalam download1music.ir (~
## 5 https://dl.download1music.ir/Music/Without-Words/Obikalam download1music.ir (~
## 6 https://dl.download1music.ir/Music/Without-Words/Obikalam download1music.ir (~
```

```
links[1:6 , 2]
```

```
## # A tibble: 6 x 1
##   file_name
##   <chr>
## 1 Obikalam download1music.ir (1).mp3
## 2 Obikalam download1music.ir (10).mp3
## 3 Obikalam download1music.ir (11).mp3
## 4 Obikalam download1music.ir (12).mp3
## 5 Obikalam download1music.ir (2).mp3
## 6 Obikalam download1music.ir (3).mp3
```

Web Scraping

Now we want to downlod our links with download.file function that the first arg is link to download and second arg is the address to save it.
the apply function here is downloading 1 to 3 first links on my dekstop.

```
apply(links[1:3, ], 1, function(x){
  try(download.file(x[1],paste0("C:/Users/Frostless/Desktop/",x[2])))
})
```

```
## [1] 0 0 0
```

End

Hey you

Thanks for youre attention