Mini Project 01 - IMDB web scraping

goal: pull the rating, metascore from imdb top 100 movies

right click > inspect on webpage

```
library(tidyverse)
library(rvest) # scrape data from the internet
```

```
Warning message in system("timedatectl", intern = TRUE):
“running command 'timedatectl' had status 1”
Warning message:
“Failed to locate timezone database”
── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 

✓ ggplot2 3.3.5     ✓ purrr   0.3.4
✓ tibble  3.1.5     ✓ dplyr   1.0.7
✓ tidyr   1.1.4     ✓ stringr 1.4.0
✓ readr   2.0.2     ✓ forcats 0.5.1

── Conflicts ────────────────────────────────────────── tidyverse_conflicts() 
✖ dplyr::filter()  masks stats::filter()
✖ purrr::flatten() masks jsonlite::flatten()
✖ dplyr::lag()     masks stats::lag()


Attaching package: ‘rvest’
```

```
url <- "https://www.imdb.com/search/title/?groups=top_100&sort=user_rating,desc"
print(url)
```

```
[1] "https://www.imdb.com/search/title/?groups=top_100&sort=user_rating,desc"
```

```
# read html
imdb <- read_html(url)
imdb
```

```
{html_document}
<html xmlns:og="http://ogp.me/ns#" xmlns:fb="http://www.facebook.com/2008/fbml"
```

```
[1] <head>\n<meta http-equiv="Content-Type" content="text/html; charset=UTF-8 .
[2] <body id="styleguide-v2" class="fixed">\n            <img height="1" widt .
```

```
# movie title
imdb %>%
    html_node("h3.lister-item-header") %>%
    html_text2()

# html_text -> includes special characters
# html_text2 -> remove special characters
```

'1. The Shawshank Redemption (1994)'

```
# multiple movie title
# use html_nodes()
titles <- imdb %>%
    html_nodes("h3.lister-item-header") %>%
    html_text2()

titles
```

'1. The Shawshank Redemption (1994)' · '2. The Godfather (1972)' · '3. The Dark Knight (2008)' ·
'4. The Godfather Part II (1974)' · '5. Schindler\'s List (1993)' · '6. 12 Angry Men (1957)' ·
'7. The Lord of the Rings: The Return of the King (2003)' · '8. Pulp Fiction (1994)' ·
'9. The Lord of the Rings: The Fellowship of the Ring (2001)' · '10. Inception (2010)' · '11. Fight Club (1999)' ·
'12. Forrest Gump (1994)' · '13. The Lord of the Rings: The Two Towers (2002)' ·
'14. Il buono, il brutto, il cattivo (1966)' · '15. GoodFellas (1990)' · '16. One Flew Over the Cuckoo\'s Nest (1975)' ·
'17. The Matrix (1999)' · '18. The Empire Strikes Back (1980)' · '19. Interstellar (2014)' ·
'20. The Silence of the Lambs (1991)' · '21. The Green Mile (1999)' · '22. Se7en (1995)' · '23. Star Wars (1977)' ·
'24. Terminator 2: Judgment Day (1991)' · '25. Sen to Chihiro no kamikakushi (2001)' ·
'26. Saving Private Ryan (1998)' · '27. Cidade de Deus (2002)' · '28. La vita è bella (1997)' ·
'29. It\'s a Wonderful Life (1946)' · '30. Shichinin no samurai (1954)' · '31. Seppuku (1962)' ·
'32. The Departed (2006)' · '33. Whiplash (2014)' · '34. Gisaengchung (2019)' · '35. Gladiator (2000)' ·
'36. Back to the Future (1985)' · '37. The Prestige (2006)' · '38. Apocalypse Now (1979)' · '39. Alien (1979)' ·
'40. Léon (1994)' · '41. The Lion King (1994)' · '42. The Usual Suspects (1995)' · '43. American History X (1998)' ·
'44. The Pianist (2002)' · '45. Casablanca (1942)' · '46. The Intouchables (2011)' · '47. Psycho (1960)' ·
'48. Once Upon a Time in the West (1968)' · '49. Hotaru no haka (1988)' · '50. Nuovo Cinema Paradiso (1988)'

```
# movie ratings
# goal -> find the smallest html block
ratings <- imdb %>%
    html_nodes("div.ratings-imdb-rating") %>%
    html_text2() %>%
    as.numeric()

ratings[1:10]
```

9.3 · 9.2 · 9 · 9 · 9 · 9 · 9 · 8.9 · 8.8 · 8.8

```
# whole vote / gross information box
num_votes <- imdb %>%
    html_nodes("p.sort-num_votes-visible") %>%
    html_text2()
num_votes[1:10]
```

'Votes: 2,705,483 | Gross: $28.34M | Top 250: #1' · 'Votes: 1,878,665 | Gross: $134.97M | Top 250: #2' · 'Votes: 2,679,081 | Gross: $534.86M | Top 250: #3' · 'Votes: 1,283,030 | Gross: $57.30M | Top 250: #4' · 'Votes: 1,367,323 | Gross: $96.90M | Top 250: #6' · 'Votes: 799,255 | Gross: $4.36M | Top 250: #5' · 'Votes: 1,862,596 | Gross: $377.85M | Top 250: #7' · 'Votes: 2,077,108 | Gross: $107.93M | Top 250: #8' · 'Votes: 1,892,040 | Gross: $315.54M | Top 250: #9' · 'Votes: 2,377,085 | Gross: $292.58M | Top 250: #14'

```
# build a dataset (final)
df <- data.frame(
    title = titles,
    rating = ratings,
    num_vote = num_votes
)
head(df)
```

A data.frame: 6 × 3

	title	rating	num_vote
	<chr>	<dbl>	<chr>
1	1. The Shawshank Redemption (1994)	9.3	Votes: 2,705,483 | Gross: $28.34M | Top 250: #1
2	2. The Godfather (1972)	9.2	Votes: 1,878,665 | Gross: $134.97M | Top 250: #2
3	3. The Dark Knight (2008)	9.0	Votes: 2,679,081 | Gross: $534.86M | Top 250: #3
4	4. The Godfather Part II (1974)	9.0	Votes: 1,283,030 | Gross: $57.30M | Top 250: #4
5	5. Schindler's List (1993)	9.0	Votes: 1,367,323 | Gross: $96.90M | Top 250: #6
6	6. 12 Angry Men (1957)	9.0	Votes: 799,255 | Gross: $4.36M | Top 250: #5

Mini Project 02 - Specphone Phone Database

goal: pull the phone specs

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

```
Warning message in system("timedatectl", intern = TRUE):
“running command 'timedatectl' had status 1”
Warning message:
“Failed to locate timezone database”
── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1

✓ ggplot2 3.3.5     ✓ purrr   0.3.4
✓ tibble  3.1.5     ✓ dplyr   1.0.7
✓ tidyr   1.1.4     ✓ stringr 1.4.0
✓ readr   2.0.2     ✓ forcats 0.5.1

── Conflicts ────────────────────────────────────────── tidyverse_conflicts()
✖ dplyr::filter()  masks stats::filter()
✖ purrr::flatten() masks jsonlite::flatten()
✖ dplyr::lag()     masks stats::lag()


Attaching package: ‘rvest’
```

```
url2 <- read_html("https://specphone.com/Samsung-Galaxy-A04.html")
print(url2)
```

```
{html_document}
<html class="no-js" lang="en-US">
[1] <head itemscope itemtype="https://schema.org/WebSite">\n<meta http-equiv= .
[2] <body id="blog" class="wp-custom-logo wp-embed-responsive main" itemscope .
```

```
att <- url2 %>%
    html_nodes("div.topic") %>%
    html_text2()
```

```
detail <- url2 %>%
    html_nodes("div.detail") %>%
    html_text2()
```

```
data.frame(
    attribute = att,
    value = detail
)
```

A data.frame: 31 × 2

attribute	value
<chr>	<chr>
วันเปิดตัว	**ตุลาคม** 2565
วันวางจำหน่าย	**ยังไม่วางจำหน่าย**
ขนาด	164.40 x 76.30 x 9.10 **มม.**
น้ำหนัก	192 **กรัม**
วัสดุ	Glass front, plastic back, plastic frame
SIM	**รองรับ** 2 **ซิมการ์ด** (nano sim, nano sim)
Technology	HSPA 42.2/5.76 Mbps, LTE-A
2G	850/900/1800/1900
3G	850/900/1900/2100
4G	850/900/1900/2100/2600
5G	-
ความเร็ว	HSPA 42.2/5.76 Mbps, LTE-A
ประเภท	PLS LCD
ขนาดหน้าจอ	6.50 **นิ้ว**
ความละเอียด	720 x 1600 pixels
ระบบปฏิบัติการ	Android 12
ชิปประมวลผล	Spreadtrum Unisoc SC9863A 1.6 GHz
ชิปกราฟิก	PowerVR GE8322
หน่วยความจำ	3 GB
ความจุ	32 GB
Memory Card	microSD (1)
กล้องหลัก	**ตัวที่** 1: 50 MP, f/1.8, (wide), AF **ตัวที่** 2: 2 MP, f/2.4, (depth)
ความละเอียดวีดีโอ	1080p@30fps
กล้องหน้า	**ตัวที่** 1: 5 MP, f/2.2
Bluetooth	5.0, A2DP, LE
Wi-Fi	802.11 a/b/g/n/ac, dual-b
USB	Type-C
GPS	GLONASS, GALILEO, BDS
NFC	**ไม่รองรับ**
ความจุ	5,000 mAh
ประเภท	Non-removable Li-Po Batt

key concept:

- request document from the website
- obtain the block

- specify the block and pull the data

```
# ALL samsung smartphones
url3 <- read_html("https://specphone.com/brand/Samsung")
url3
```

```
{html_document}
<html class="no-js" lang="en-US">
[1] <head itemscope itemtype="https://schema.org/WebSite">\n<meta http-equiv= .
[2] <body id="blog" class="wp-custom-logo wp-embed-responsive main" itemscope .
```

```
# need to  know each link of the model
# goal: get href
links <- url3 %>%
    html_nodes("li.mobile-brand-item a") %>%
    html_attr("href")

links
# need to klnow css stuffs
# a after the block name = child name : find this attribute that has "a"
```

'/Samsung-Galaxy-M13.html' · '/Samsung-Galaxy-A23.html' · '/Samsung-Galaxy-A13.html' · '/Samsung-Galaxy-M32-5G.html' · '/Samsung-Galaxy-A12-Nacho.html' · '/Samsung-Galaxy-Pocket-Neo.html' · '/Samsung-Galaxy-Young.html' · '/Samsung-Galaxy-J1-Mini.html' · '/Samsung-Galaxy-A01-Core-1-16GB.html' · '/Samsung-Galaxy-V-PLUS.html' · '/Samsung-Galaxy-Young-2.html' · '/Samsung-Galaxy-M02.html' · '/Samsung-Galaxy-A11.html' · '/Samsung-Galaxy-J2-Pro-2018.html' · '/Samsung-Galaxy-A12-2021.html' · '/Samsung-Galaxy-A21s-3-32GB.html' · '/Samsung-Galaxy-J5.html' · '/Samsung-Galaxy-J4.html' · '/Samsung-Galaxy-Core-2-Duos.html' · '/Samsung-Galaxy-Ace-Plus.html' · '/Samsung-Galaxy-A20.html' · '/Samsung-Galaxy-Chat.html' · '/Samsung-Galaxy-Gio.html' · '/Samsung-Galaxy-Tab-A7-Lite-LTE.html' · '/Samsung-Galaxy-Tab-A-10.5WIFI.html' · '/Samsung-Galaxy-Alpha.html' · '/Samsung-Galaxy-S3-Slim.html' · '/Samsung-Galaxy-S4-zoom.html' · '/Samsung-Galaxy-Xcover-2.html' · '/Samsung-Galaxy-Tab-8.9-3G-16GB.html' · '/Samsung-Galaxy-Tab-A8-LTE-2021.html' · '/Samsung-Galaxy-A8-2018.html' · '/Samsung-Galaxy-Tab4-8.0-wifi.html' · '/Samsung-Galaxy-M33-5G.html' · '/Samsung-Galaxy-A50.html' · '/Samsung-Galaxy-E7.html' · '/Samsung-Galaxy-S6.html' · '/Samsung-Galaxy-S20-FE.html' · '/Samsung-Galaxy-Tab-S4-WIFI.html' · '/Samsung-Galaxy-S7.html' · '/Samsung-Galaxy-Note-5-Exynos.html' · '/Samsung-Galaxy-TabPRO-12.2-LTE.html' · '/Samsung-Galaxy-S4-Active.html' · '/Samsung-Galaxy-Tab-Active-3.html' · '/Samsung-Galaxy-Tab-S3-9.7.html' · '/Samsung-Galaxy-S6-edge.html' · '/Samsung-Galaxy-Note-4-Exynos.html' · '/Samsung-Galaxy-Round.html' · '/Samsung-Galaxy-Note-20-Ultra-5G.html' · '/Samsung-ATIV-Q.html' · '/Samsung-ATIV-Smart-PC-PRO.html' · '/Samsung-Galaxy-S23-Ultra-5G8-256GB.html' · '/Samsung-Galaxy-S22-Ultra12-128GB.html' · '/Samsung-Galaxy-Z-Flip-5G.html' · '/Samsung-Galaxy-Z-Flip.html' · '/Samsung-Galaxy-Tab-S8-Ultra-5G.html' · '/Samsung-Galaxy-S21-Ultra-16-512GB.html' · '/Samsung-Galaxy-S23-Ultra-5G.html' · '/Samsung-Galaxy-S10-Plus-Ram-12GB.html' · '/Samsung-Galaxy-Z-Fold-3.html'

```
full_links <- paste0("https://specphone.com", links)
```

```
result <- data.frame()

for (link in full_links[1:10]){
    ss_topic <- link %>%
        read_html() %>%
        html_nodes("div.topic") %>%
        html_text2()

    ss_detail <- link %>%
        read_html() %>%
        html_nodes("div.detail") %>%
        html_text2()
    tmp <- data.frame(attribute = ss_topic,
                      value = ss_detail)
    result <- bind_rows(result,tmp)
    print("progress...")
}

print(result)
```

```
[1] "progress..."
[1] "progress..."
[1] "progress..."
[1] "progress..."
[1] "progress..."
[1] "progress..."
[1] "progress..."
[1] "progress..."
[1] "progress..."
[1] "progress..."
         attribute
1          วันเปิดตัว
2      วันวางจำหน่าย
3              ขนาด
4            น้ำหนัก
5              วัสดุ
6              SIM
7       Technology
8               2G
9               3G
```

```
# write csv
write_csv(result, "result_ss_phone.csv")
```

```
print(head(result))
```

```
    attribute                                        value
1     วันเปิดตัว                                 มิถุนายน 2565
2 วันวางจำหน่าย                               ยังไม่วางจำหน่าย
3        ขนาด                  165.40 x 76.90 x 8.40 มม.
4       น้ำหนัก                                      192 กรัม
5         วัสดุ Glass front, plastic back, plastic frame
6         SIM      รองรับ 2 ซิมการ์ด (nano sim, nano sim)
```