RWorksheet#5

LANGREO,LIZA,PABRIAGA,SISON

2024-11-12

```
options(repos = c(CRAN = "https://cloud.r-project.org/"))

#1_Shoes
# Install and load the rvest package
#if (!requireNamespace("rvest", quietly = TRUE)) {
  install.packages("rvest")
```

```
##
## The downloaded binary packages are in
##  /var/folders/w4/vbll1gqj0dx2bgx96nx57rp80000gn/T//RtmptILTjF/downloaded_packages
```

```
#}
library(rvest)
library(polite)

# Read the HTML file
url <- "https://www.amazon.com/s?k=amazon+men%27s+shoes&language=en_US&adgrpid=142537954933&hvadid=6735
```

```
Ses<- bow(url, user_agent = "Student's Demo Educational")
Ses
```

```
## <polite session> https://www.amazon.com/s?k=amazon+men%27s+shoes&language=en_US&adgrpid=142537954933
##     User-agent: Student's Demo Educational
##     robots.txt: 138 rules are defined for 5 bots
##    Crawl delay: 5 sec
##   The path is scrapable for this user-agent
```

```
Page <- scrape(Ses)

# Find all div elements with the specified class
div<- html_nodes(Page,'div.sg-col-4-of-24.sg-col-4-of-12.s-result-item.s-asin.sg-col-4-of-16.sg-col.s-w

# Create empty vectors to store data
links <- character()
img_srcs <- character()
titles <- character()
prices <- character()
ratings <- character()

max_prod<- 30

# Limit the loop to only collect data for the first 30 products
for (i in 1:min(length(div), max_prod)) {
  div_e <- div[i]
```

```r
  # Find the a element with class="a-link-normal s-no-outline" and get the link
  a_element <- html_node(div_e, 'a.a-link-normal.s-no-outline')
  link <- ifelse(!is.na(a_element), paste0("https://amazon.com", html_attr(a_element, "href")), '')

  # Find the img element with class="s-image" and get the link
  img <- html_node(div_e, 'img.s-image')
  img_src <- ifelse(!is.na(img), html_attr(img , "src"), '')

  # Find the span element with class="a-size-base-plus a-color-base a-text-normal" and get the title
  title_e<- html_node(div_e, 'span.a-size-base-plus.a-color-base.a-text-normal')
  title <- ifelse(!is.na(title_e), html_text(title_e), '')

  # Find the span element with class="a-price-whole" and get the price
  price_e<- html_node(div_e, 'span.a-price-whole')
  price <- ifelse(!is.na(price_e), html_text(price_e), '')

  # Find the span element with class="a-icon-alt" and get the ratings
  rating_e<- html_node(div_e, 'span.a-icon-alt')
  rating <- ifelse(!is.na(rating_e), html_text(rating_e), '')

  # Append data to vectors
  links <- c(links, link)
  img_srcs <- c(img_srcs, img_src)
  titles <- c(titles, title)
  prices <- c(prices, price)
  ratings <- c(ratings, rating)
}

# Create a data frame with the scraped data
prod_df <- data.frame(
  Links = links,
  Images = img_srcs,
  Title = titles,
  Price = prices,
  Rating = ratings
)

# Write the data to a CSV file
write.csv(prod_df, "Shoesable.csv", row.names = FALSE)
```

```r
options(repos = c(CRAN = "https://cloud.r-project.org/"))

#2_Bags
# Install and load the rvest package
#if (!requireNamespace("rvest", quietly = TRUE)) {
  install.packages("rvest")
```

```
##
## The downloaded binary packages are in
##  /var/folders/w4/vbll1gqj0dx2bgx96nx57rp80000gn/T//RtmptILTjF/downloaded_packages
```

```r
#}
library(rvest)
library(polite)
```

```r
# Read the HTML file
url <- "https://www.amazon.com/bag/s?k=bag"

Ses<- bow(url, user_agent = "Student's Demo Educational")
Ses
```

```
## <polite session> https://www.amazon.com/bag/s?k=bag
##     User-agent: Student's Demo Educational
##     robots.txt: 138 rules are defined for 5 bots
##    Crawl delay: 5 sec
##   The path is scrapable for this user-agent
```

```r
Page <- scrape(Ses)

# Find all div elements with the specified class
div<- html_nodes(Page,'div.sg-col-4-of-24.sg-col-4-of-12.s-result-item.s-asin.sg-col-4-of-16.sg-col.s-w

# Create empty vectors to store data
links <- character()
img_srcs <- character()
titles <- character()
prices <- character()
ratings <- character()

max_prod<- 30

# Limit the loop to only collect data for the first 30 products
for (i in 1:min(length(div), max_prod)) {
  div_e <- div[i]

  # Find the a element with class="a-link-normal s-no-outline" and get the link
  a_element <- html_node(div_e, 'a.a-link-normal.s-no-outline')
  link <- ifelse(!is.na(a_element), paste0("https://amazon.com", html_attr(a_element, "href")), '')

  # Find the img element with class="s-image" and get the link
  img <- html_node(div_e, 'img.s-image')
  img_src <- ifelse(!is.na(img), html_attr(img , "src"), '')

  # Find the span element with class="a-size-base-plus a-color-base a-text-normal" and get the title
  title_e<- html_node(div_e, 'span.a-size-base-plus.a-color-base.a-text-normal')
  title <- ifelse(!is.na(title_e), html_text(title_e), '')

  # Find the span element with class="a-price-whole" and get the price
  price_e<- html_node(div_e, 'span.a-price-whole')
  price <- ifelse(!is.na(price_e), html_text(price_e), '')

  # Find the span element with class="a-icon-alt" and get the ratings
  rating_e<- html_node(div_e, 'span.a-icon-alt')
  rating <- ifelse(!is.na(rating_e), html_text(rating_e), '')

  # Append data to vectors
  links <- c(links, link)
  img_srcs <- c(img_srcs, img_src)
  titles <- c(titles, title)
```

```
  prices <- c(prices, price)
  ratings <- c(ratings, rating)
}

# Create a data frame with the scraped data
prod_df <- data.frame(
  Links = links,
  Images = img_srcs,
  Title = titles,
  Price = prices,
  Rating = ratings
)

# Write the data to a CSV file
write.csv(prod_df, "Bags.csv", row.names = FALSE)
```

```
options(repos = c(CRAN = "https://cloud.r-project.org/"))

#3_Furnitures
# Install and load the rvest package
#if (!requireNamespace("rvest", quietly = TRUE)) {
  install.packages("rvest")
```

```
##
## The downloaded binary packages are in
##  /var/folders/w4/vbll1gqj0dx2bgx96nx57rp80000gn/T//RtmptILTjF/downloaded_packages
```

```
#}
library(rvest)
library(polite)

# Read the HTML file
url <-"https://www.amazon.com/s?k=furniture&crid=3IIPALF9Q6F9X&sprefix=fu%2Caps%2C482&ref=nb_sb_ss_ts-d

Ses<- bow(url, user_agent = "Student's Demo Educational")
Ses
```

```
## <polite session> https://www.amazon.com/s?k=furniture&crid=3IIPALF9Q6F9X&sprefix=fu%2Caps%2C482&ref=
##     User-agent: Student's Demo Educational
##     robots.txt: 138 rules are defined for 5 bots
##    Crawl delay: 5 sec
##   The path is scrapable for this user-agent
```

```
Page <- scrape(Ses)

# Find all div elements with the specified class
div<- html_nodes(Page,'div.sg-col-4-of-24.sg-col-4-of-12.s-result-item.s-asin.sg-col-4-of-16.sg-col.s-w

# Create empty vectors to store data
links <- character()
img_srcs <- character()
titles <- character()
prices <- character()
ratings <- character()
```

```r
max_prod<- 30

# Limit the loop to only collect data for the first 30 products
for (i in 1:min(length(div), max_prod)) {
  div_e <- div[i]

  # Find the a element with class="a-link-normal s-no-outline" and get the link
  a_element <- html_node(div_e, 'a.a-link-normal.s-no-outline')
  link <- ifelse(!is.na(a_element), paste0("https://amazon.com", html_attr(a_element, "href")), '')

  # Find the img element with class="s-image" and get the link
  img <- html_node(div_e, 'img.s-image')
  img_src <- ifelse(!is.na(img), html_attr(img , "src"), '')

  # Find the span element with class="a-size-base-plus a-color-base a-text-normal" and get the title
  title_e<- html_node(div_e, 'span.a-size-base-plus.a-color-base.a-text-normal')
  title <- ifelse(!is.na(title_e), html_text(title_e), '')

  # Find the span element with class="a-price-whole" and get the price
  price_e<- html_node(div_e, 'span.a-price-whole')
  price <- ifelse(!is.na(price_e), html_text(price_e), '')

  # Find the span element with class="a-icon-alt" and get the ratings
  rating_e<- html_node(div_e, 'span.a-icon-alt')
  rating <- ifelse(!is.na(rating_e), html_text(rating_e), '')

  # Append data to vectors
  links <- c(links, link)
  img_srcs <- c(img_srcs, img_src)
  titles <- c(titles, title)
  prices <- c(prices, price)
  ratings <- c(ratings, rating)
}

# Create a data frame with the scraped data
prod_df <- data.frame(
  Links = links,
  Images = img_srcs,
  Title = titles,
  Price = prices,
  Rating = ratings
)

# Write the data to a CSV file
write.csv(prod_df, "Furnitures.csv", row.names = FALSE)
```

```r
options(repos = c(CRAN = "https://cloud.r-project.org/"))

#4_Clothes
# Install and load the rvest package
#if (!requireNamespace("rvest", quietly = TRUE)) {
  install.packages("rvest")
```

```
##
```

```
## The downloaded binary packages are in
##  /var/folders/w4/vbll1gqj0dx2bgx96nx57rp80000gn/T//RtmptILTjF/downloaded_packages
```

```
#}
library(rvest)
library(polite)

# Read the HTML file
url <- "https://www.amazon.com/clothes/s?k=clothes"

Ses<- bow(url, user_agent = "Student's Demo Educational")
Ses
```

```
## <polite session> https://www.amazon.com/clothes/s?k=clothes
##     User-agent: Student's Demo Educational
##     robots.txt: 138 rules are defined for 5 bots
##    Crawl delay: 5 sec
##   The path is scrapable for this user-agent
```

```
Page <- scrape(Ses)

# Find all div elements with the specified class
div<- html_nodes(Page,'div.sg-col-4-of-24.sg-col-4-of-12.s-result-item.s-asin.sg-col-4-of-16.sg-col.s-w

# Create empty vectors to store data
links <- character()
img_srcs <- character()
titles <- character()
prices <- character()
ratings <- character()

max_prod<- 30

# Limit the loop to only collect data for the first 30 products
for (i in 1:min(length(div), max_prod)) {
  div_e <- div[i]

  # Find the a element with class="a-link-normal s-no-outline" and get the link
  a_element <- html_node(div_e, 'a.a-link-normal.s-no-outline')
  link <- ifelse(!is.na(a_element), paste0("https://amazon.com", html_attr(a_element, "href")), '')

  # Find the img element with class="s-image" and get the link
  img <- html_node(div_e, 'img.s-image')
  img_src <- ifelse(!is.na(img), html_attr(img , "src"), '')

  # Find the span element with class="a-size-base-plus a-color-base a-text-normal" and get the title
  title_e<- html_node(div_e, 'span.a-size-base-plus.a-color-base.a-text-normal')
  title <- ifelse(!is.na(title_e), html_text(title_e), '')

  # Find the span element with class="a-price-whole" and get the price
  price_e<- html_node(div_e, 'span.a-price-whole')
  price <- ifelse(!is.na(price_e), html_text(price_e), '')

  # Find the span element with class="a-icon-alt" and get the ratings
  rating_e<- html_node(div_e, 'span.a-icon-alt')
```

```
  rating <- ifelse(!is.na(rating_e), html_text(rating_e), '')

  # Append data to vectors
  links <- c(links, link)
  img_srcs <- c(img_srcs, img_src)
  titles <- c(titles, title)
  prices <- c(prices, price)
  ratings <- c(ratings, rating)
}

# Create a data frame with the scraped data
prod_df <- data.frame(
  Links = links,
  Images = img_srcs,
  Title = titles,
  Price = prices,
  Rating = ratings
)

# Write the data to a CSV file
write.csv(prod_df, "Clothes.csv", row.names = FALSE)
```

```
options(repos = c(CRAN = "https://cloud.r-project.org/"))

#5_School_Supplies
# Install and load the rvest package
#if (!requireNamespace("rvest", quietly = TRUE)) {
  install.packages("rvest")
```

```
##
## The downloaded binary packages are in
##  /var/folders/w4/vbll1gqj0dx2bgx96nx57rp80000gn/T//RtmptILTjF/downloaded_packages
```

```
#}
library(rvest)
library(polite)

# Read the HTML file
url <- "https://www.amazon.com/school-supplies/s?k=school+supplies"

Ses<- bow(url, user_agent = "Student's Demo Educational")
Ses
```

```
## <polite session> https://www.amazon.com/school-supplies/s?k=school+supplies
##     User-agent: Student's Demo Educational
##     robots.txt: 138 rules are defined for 5 bots
##    Crawl delay: 5 sec
##   The path is scrapable for this user-agent
```

```
Page <- scrape(Ses)

# Find all div elements with the specified class
div<- html_nodes(Page,'div.sg-col-4-of-24.sg-col-4-of-12.s-result-item.s-asin.sg-col-4-of-16.sg-col.s-w

# Create empty vectors to store data
```

```r
links <- character()
img_srcs <- character()
titles <- character()
prices <- character()
ratings <- character()

max_prod<- 30

# Limit the loop to only collect data for the first 30 products
for (i in 1:min(length(div), max_prod)) {
  div_e <- div[i]

  # Find the a element with class="a-link-normal s-no-outline" and get the link
  a_element <- html_node(div_e, 'a.a-link-normal.s-no-outline')
  link <- ifelse(!is.na(a_element), paste0("https://amazon.com", html_attr(a_element, "href")), '')

  # Find the img element with class="s-image" and get the link
  img <- html_node(div_e, 'img.s-image')
  img_src <- ifelse(!is.na(img), html_attr(img , "src"), '')

  # Find the span element with class="a-size-base-plus a-color-base a-text-normal" and get the title
  title_e<- html_node(div_e, 'span.a-size-base-plus.a-color-base.a-text-normal')
  title <- ifelse(!is.na(title_e), html_text(title_e), '')

  # Find the span element with class="a-price-whole" and get the price
  price_e<- html_node(div_e, 'span.a-price-whole')
  price <- ifelse(!is.na(price_e), html_text(price_e), '')

  # Find the span element with class="a-icon-alt" and get the ratings
  rating_e<- html_node(div_e, 'span.a-icon-alt')
  rating <- ifelse(!is.na(rating_e), html_text(rating_e), '')

  # Append data to vectors
  links <- c(links, link)
  img_srcs <- c(img_srcs, img_src)
  titles <- c(titles, title)
  prices <- c(prices, price)
  ratings <- c(ratings, rating)
}

# Create a data frame with the scraped data
prod_df <- data.frame(
  Links = links,
  Images = img_srcs,
  Title = titles,
  Price = prices,
  Rating = ratings
)

# Write the data to a CSV file
write.csv(prod_df, "School_Supplies.csv", row.names = FALSE)
```