webscraping

14 February, 2023

Write the R code to answer the following questions. You have until the beginning of next class to answer all of the questions below and commit to GitHub, both the .Rmd file and the .pdf.

Overview

Our goal is to scrape the office hours of all Notre Dame political science faculty. This activity will ask you to nagivate across webpages, find the information of interest within each page, and store it in a dataframe.

While our goal for now is to scrape just the office hours of the faculty, note that this week's problem set will ask you scrape more of the listed details on each page and organize them into a dataframe.

Question 1

Our first step is to be able to navigate to each faculty member's page. Write code to extract the links for each faculty member's page from the provided html. Print your vector of strings. There should be 50.

```
library(rvest)
url <- "https://politicalscience.nd.edu/people/core-faculty/"
html <- read_html(url)

# notice there are 3 for each person!
all_page_links <- html %>% html_nodes('a') %>%
  html_attr('href')

# better approach would be to get links
# through image or name tag, specifically
faculty_links <- html %>%
  html_nodes('h2.faculty-item-name') %>%
  html_nodes('a') %>%
  html_attr('href')
faculty_links
```

```
##  [1] "/people/christina-bambrick/"    "/people/sotirios-barber/"
##  [3] "/people/jaimie-bleck/"          "/people/david-campbell/"
##  [5] "/people/susan-d-collins/"       "/people/michael-j-coppedge/"
##  [7] "/people/david-cortez/"          "/people/darren-davis/"
##  [9] "/people/patrick-j-deneen/"      "/people/michael-c-desch/"
## [11] "/people/rev-robert-dowd-c-s-c/" "/people/amitava-krishna-dutt/"
## [13] "/people/luis-ricardo-fraga/"    "/people/eugene-gholz/"
## [15] "/people/andrew-c-gould/"        "/people/matthew-e-k-hall/"
## [17] "/people/jeff-harden/"           "/people/michael-hoffman/"
## [19] "/people/victoria-tin-bor-hui/"  "/people/eileen-m-hunt/"
## [21] "/people/debra-javeline/"        "/people/joshua-b-kaplan/"
## [23] "/people/rosemary-a-kelanic/"    "/people/mary-m-keys/"
## [25] "/people/karrie-j-koesel/"       "/people/geoffrey-c-layman/"
```

```
## [27] "/people/dan-lindley/"            "/people/scott-mainwaring/"
## [29] "/people/a-james-mcadams/"        "/people/angela-mccarthy/"
## [31] "/people/vincent-phillip-munoz/" "/people/joseph-m-parent/"
## [33] "/people/anibal-perez-linan/"     "/people/daniel-philpott/"
## [35] "/people/dianne-pinderhughes/"    "/people/rachel-porter/"
## [37] "/people/emilia-justyna-powell/" "/people/benjamin-radcliff/"
## [39] "/people/ricardo-ramirez/"        "/people/luc-reydams/"
## [41] "/people/susan-pratt-rosato/"     "/people/sebastian-rosato/"
## [43] "/people/erin-rossiter/"          "/people/luis-schiumerini/"
## [45] "/people/jazmin-sierra/"          "/people/guillermo-trejo/"
## [47] "/people/ernesto-verdeja/"        "/people/dana-villa/"
## [49] "/people/susanne-wengle/"         "/people/christina-wolbrecht/"
```

```
length(faculty_links)
```

```
## [1] 50
```

Question 2

Investigate these links and how the website is structured. Before writing any code, briefly describe in words the kind of code you'll write to accomplish our goal of navigating to each faculty member's page, selecting the office hours information, and saving it to a dataframe.

Question 3

Implement the code you described in words in Question 2. I've provided the structure of a dataframe in which I'd like you to add the office hours information.

```
base_url <- "https://politicalscience.nd.edu"
faculty_df <- data.frame("link" = faculty_links,
                         "office_hours" = NA)

# iterate by index
for(i in 1:nrow(faculty_df)){
  # paste the faculty extension to the base URL
  url <- paste0(base_url, faculty_df$link[i])
  # get html
  html <- read_html(url)
  # get just the "faculty details" div's
  p_tags <- html %>%
    html_nodes('div.faculty-details') %>%
    html_nodes('p') %>%
    html_text2()
  # use our text skills to find office hours element
  idx <- which(grepl("Office Hours", p_tags))
  # it might be missing! if so, next iteration
  # if not, store it in the dataframe
  if(length(idx) == 0){
    next
  }else{
    faculty_df$office_hours[i] <- p_tags[idx]
  }
}
write.csv(faculty_df, file = "faculty_df_answer.csv")
```