[09:00 ~ 10:50]

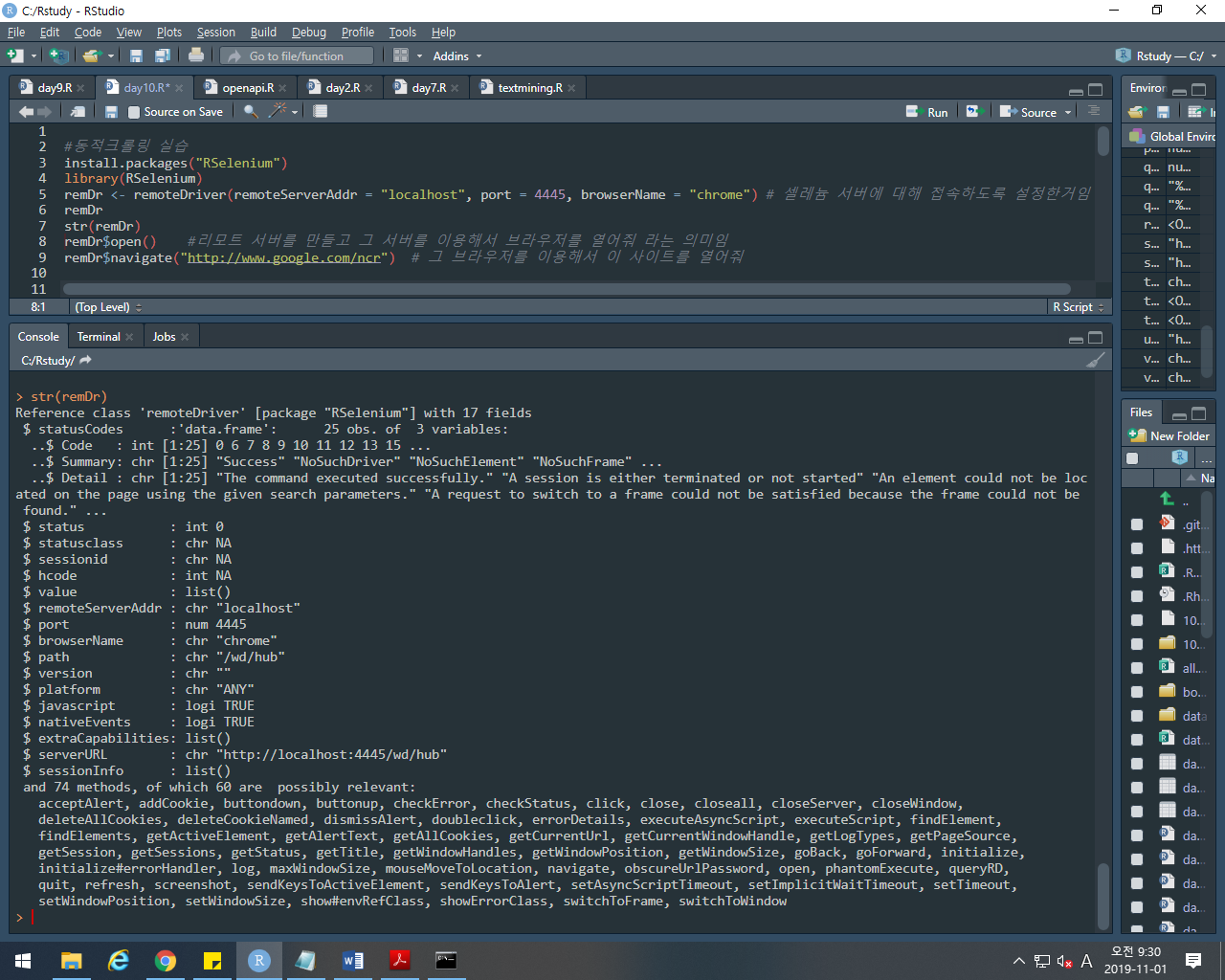
@동적크롤링 어제 배운내용 복습

1. 셀레니엄 가동 코드 분석

remDr <- remoteDriver(remoteServerAddr = "localhost", port = 4445, browserName = "chrome")

localhost는 지금 사용하는 컴퓨터를 의미함, 포트는 어디에 있는 서버로 접속할 것인가, 브라우저 이름은 사용할 브라우저를 지정하는 것 (그냥 저대로 적으면됨.)

remDr, str(remDr)해보면 list의 형식을 띄고 있는 것을 알 수 있음



<str실행한 화면임>

파란색으로 표시한 부분이 method임, 함수랑 기본적으로 기능은 같지만 함수는 개별적으로 호출 할 수 있는 대신에 method는 “상위주체$method”로 사용가능하다.

(ex remDr$open remDr이 상위 부분, $ 뒤 open이 method임 총 74개의 method가 있음)

1. webElem의 속성 보기

webElem <- remDr$findElement(using = "css", "[name = 'q']")

webElem

str(webElem)

webElem$sendKeysToElement(list("JAVA",key = "enter"))

webElem$getElementTagName()

간혹 뽑아온 elements의 속성을 봐야 하는 경우가 있음 그 경우 마지막 명령어를 수행한 다(정확히 우리가 사용은 안했음)

1. 네이버 웹툰 예시

베스트댓글 긁어오기

bestReviewNodes <- remDr$findElements(using = "css selector", "div.u\_cbox\_text\_wrap > span.u\_cbox\_contents")

sapply(bestReviewNodes, function(x){x$getElementText()})

전체 댓글 링크 클릭후에 첫 페이지 내용 읽어오기

totalReview <- remDr$findElement(using = "css", "span.u\_cbox\_sort\_label")

totalReview$clickElement()

totalReviewNodes <- remDr$findElements(using = "css selector", "div > div.u\_cbox\_text\_wrap > span")

sapply(totalReviewNodes,function(x){x$getElementText()})

AJAX통신은 비동기통신임. 비동기통신이란 앞서 신청한 정보가 와서 렌더링 하기 전에 앞단계의 명령을 먼저 수행하는 것임. 그래서 가끔 정보를 불러온 다음에 다음 수행을 해야되는데 네트워크 상태로 인하여 전 단계가 지연되어도 앞서 일을 진행하려하기 때문에 오류가 나는 경우가 있음. 이 경우에는 페이지 이동 시 Sys.sleep(1)정도를 줘서 텀을 주는 것이 유용할 수 있다.

\*css와 css selector 차이점

Css가 더 상위의 개념임 css selector 와 css attribute로 구성되어 있음. 엄밀히 말하면 우리가 찾아서 쓰는 것은 css selector임. 그런데 using에서는 css와 css selector를 구분안해줘도 다 selector로 인식함.

\*브라우저가 랜더링하지 않은 정보들은 읽어올 수 없음. 즉, 셀레니움이 수행을 하도록 만든다음에 정보들을 가져올 수 있음. (AJAX를 수행하여 끌어오게 해야됨.) (앞서말한 지연시간 1초 주는 이유.)

[11:00 ~ 11:50]

@실습 : 네이버 웹툰 댓글에서 BEST댓글 + 전체 댓글 10페이지를 읽어서 webtoon1.txt에 저장

순서

1. 네이버 웹툰 페이지 네비게이팅
2. 베스트댓글을 읽어서 변수 저장
3. 전체댓글 보기 링크 클릭
4. 첫페이지 댓글내용 읽고 변수에 추가
5. 2 ~ 10페이지까지 이동하여 댓글내용 읽어서 변수에 추가
6. 결과물을 webtoon1.txt에 저장하여 webtoon1.R과 함께 제출

제출 후 공유폴더에서 전체댓글 끌어오는 소스 받아서 리뷰해보기

[13:00 ~ 15:50]

아고다 – 신라스테이 리뷰 읽어오기 실습

나중에 확인 클릭하여 페이지 활성화 시키기.

test <- NULL

test <- remDr$findElement(using = "css", "#customer-reviews-panel > span")

test

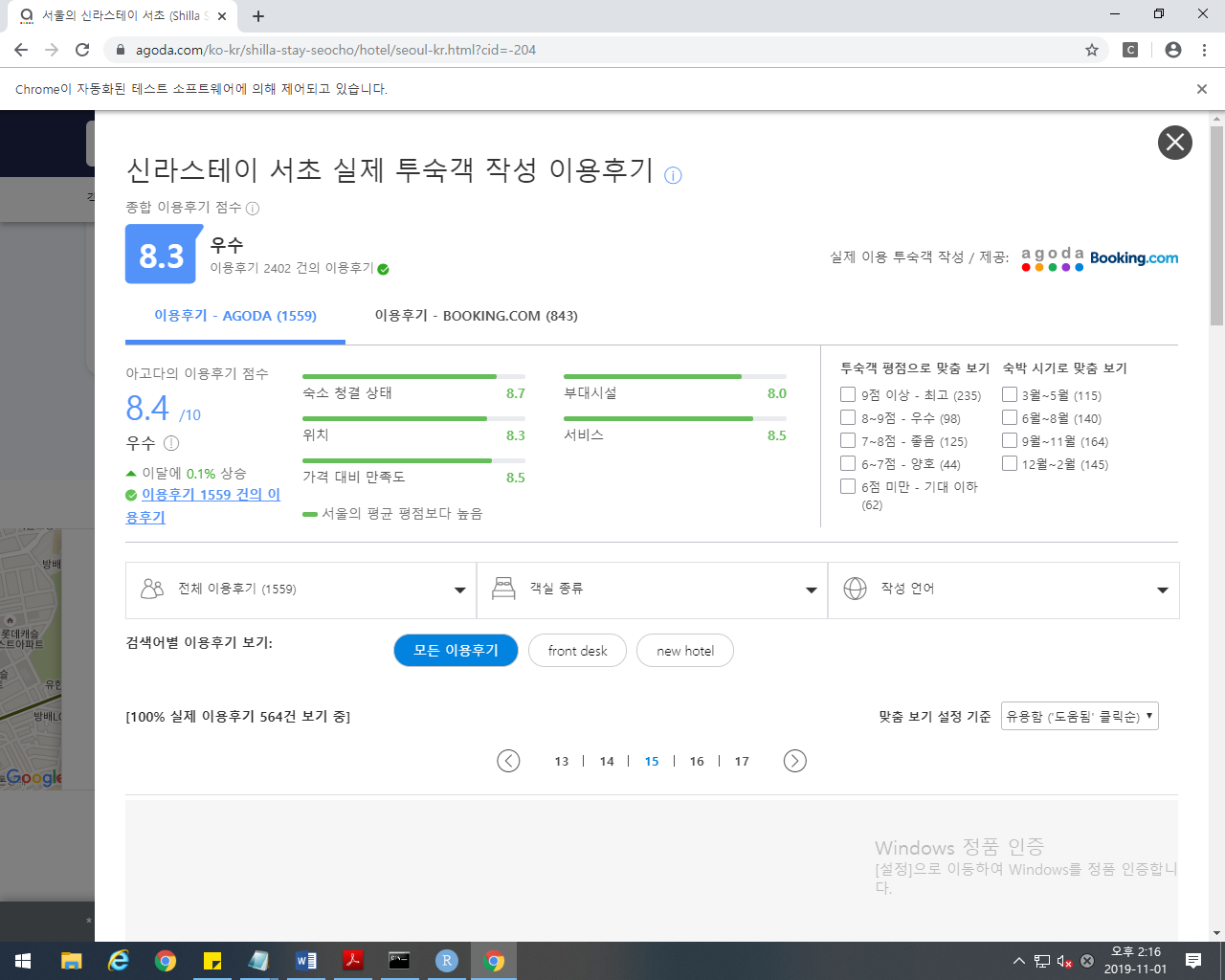
입력하여 test결과 나오는지 확인해보기 (안나와야 정상)

왜냐면 스크롤링을 해야 밑에 내용들을 불러오기 때문임

remDr$executeScript("scrollBy(0, 8400)")

이 소스 입력하면 스크롤하는 결과가 나타남

신라스테이 이용후기를 눌러서 페이지를 넘겨보기



페이지를 넘길때마다 숫자가 변하기 때문에 (네이버 웹툰하고 다르게) 페이지넘기는 방법을 다르게 설정해야함 -> 다음페이지로 넘어가는 >버튼 누르거나 1,2,3페이지 까지 간 다음 4번째 숫자만 계속누르고 마지막페이지 에가서는 5번째 가기

@실습2 아고다 호텔후기 끝까지 크롤링하기

공유폴더에 선생님 작성답안 있음. 확인할것!

[16:00 ~ 16:50]

Yes24실습 하기

리뷰를 가져오려면 스크롤을 해야 리뷰가 나타나게됨. 적당히 잘 내려가야됨 스크롤다운을

게다가 yes24리뷰는 짧은댓글과 긴댓글이 존재함. (펼쳐보기를 알아서 하나씩 클릭해야함)

scrollTo와 scrollBy가 있는데 To는 왼쪽위 기준점을 중심으로 절대값을 의미하고 By는 현재에서 부터임

# [ YES24의 명견만리 댓글 읽어오기 ] 소스 리뷰하기

library(RSelenium)

remDr <- remoteDriver(remoteServerAddr = "localhost", port = 4445, browserName = "chrome")

remDr$open()

remDr$navigate("http://www.yes24.com/24/goods/40936880")

webElem <- remDr$findElement("css", "body")

remDr$executeScript("scrollTo(0, 0)", args = list(webElem))

Sys.sleep(1)

remDr$executeScript("scrollBy(0, 2500)", args = list(webElem))

Sys.sleep(1)

remDr$executeScript("scrollBy(0, 2500)", args = list(webElem))

Sys.sleep(1)

remDr$executeScript("scrollBy(0, 1000)", args = list(webElem))

Sys.sleep(3)

repl\_v = NULL

endFlag <- FALSE

page <- 3

repeat {

for(index in 3:7) {

fullContentLinkCSS <- paste("#infoset\_reviewContentList > div:nth-child(",index,") > div.btn\_halfMore > a", sep='')

fullContentLink<-remDr$findElements(using='css selector', fullContentLinkCSS)

if (length(fullContentLink) == 0) {

endFlag <- TRUE

break

}

sapply(fullContentLink,function(x){x$clickElement()})

Sys.sleep(1)

fullContentCSS <- paste("div:nth-child(",index,") > div.reviewInfoBot.origin > div.review\_cont > p", sep='')

fullContent<-remDr$findElements(using='css selector', fullContentCSS)

print(fullContent)

repl <-sapply(fullContent,function(x){x$getElementText()})

print(repl)

repl\_v <- c(repl\_v, unlist(repl))

}

if(endFlag)

break;

if(page == 10){

page <- 3

nextPageCSS <- "#infoset\_reviewContentList > div.review\_sort.sortTop > div.review\_sortLft > div > a.bgYUI.next"

}

else{

page <- page+1;

nextPageCSS <- paste("#infoset\_reviewContentList > div.review\_sort.sortTop > div.review\_sortLft > div > a:nth-child(",page,")",sep="")

}

remDr$executeScript("scrollTo(0, 0)", args = list(webElem))

nextPageLink<-remDr$findElements(using='css selector',nextPageCSS)

sapply(nextPageLink,function(x){x$clickElement()})

Sys.sleep(5)

print(page)

}

write(repl\_v, "yes24.txt")

[17:00 ~ 17:50]

스타벅스 매장 이름 및 위치 찾기 실습(11월1일 2번째 실습파일)

제출을 저도 아직 못했어요…너무어려워서….

나중에 같이 설명듣죠