13. 웹 데이터 수집(웹 크롤링) # src: 7_R/13_웹데이터수집(웹크롤링) 참고

1) 정적 웹크롤링; 웹 페이지 HTML의 class명, 선택자를 이용해서 웹 크롤링

(1) 단일 페이지 크롤링(**rvest** 패키지 사용)

```
text <-read_html(url, encoding = 'CP949')
nodes <-html_nodes(text, '.클래스명')
title <-html_text(nodes)
movieInfo <-html_attr(nodes, 'href')
movieInfo <-paste0(url, movieInfo)          #paste0; space 없이 붙여줌
```

(2) 여러 페이지 정적 웹크롤링: for문, rbind() 이용

2) 동적 웹크롤링; 스크롤 다운, 로그인 이후, 버튼 클릭 등

(1) 동적 웹크롤링 준비 #selenium 패키지 이용

① 폴더 생성 후 3개 파일 다운 받아 selenium 서버 가동

② 필요한 패키지 다운로드 및 로드: RSelenium, httr, rvest

(2) 특정 부분에 text 입력한 후 엔터한 결과를 크롤링 #검색창에 검색 후 결과 크롤링

```
remDr <- remoteDriver(port=4445L, browserName='chrome') #포트 번호 및 사용할 브라우저
remDr$open()              #브라우저 창 열림
remDr$navigate('URL')     #해당 URL로 이동
webElement <- remDr$findElement(using='css', '#클래스명')
webElement$sendKeysToElement(list('검색할 내용', key='enter'))
html <- remDr$getPageSource()[[1]]          #현재 페이지의 html소스 가져오기
html <- read_html(html)                     #현재 페이지의 html소스 가져오기
title <- html %>% html_nodes('#클래스명') %>% html_text()  #해당 클래스 css의 text 가져오기
title <- gsub('₩n', '', title)              #필요없는 문자들 정리
title <- trimws(title)                      #필요없는 여백 정리
url <- html %>% html_nodes('#클래스명') %>% html_attr('href')  #url 링크 가져오기
url <-ifelse(is.na(url), '', paste0('브라우저 주소', 'url'))
result <-cbind(title, url)                  #내용 합치기
write.csv(result, file="파일명.csv", row.names=F)  #파일로 저장
```

(3) 마우스 스크롤 다운하여 크롤링: remDr$executeScript("window.scrollTo(num1, num2)") ┐

- 메인 동영상 플레이 멈추기 #y축(num2)가 커질수록 스크롤 다운

```
btn <-remDr$findElement(using='css selector', value='클래스명');  btn$clickElement();
```