**수행평가 1차**

**<Comment>**

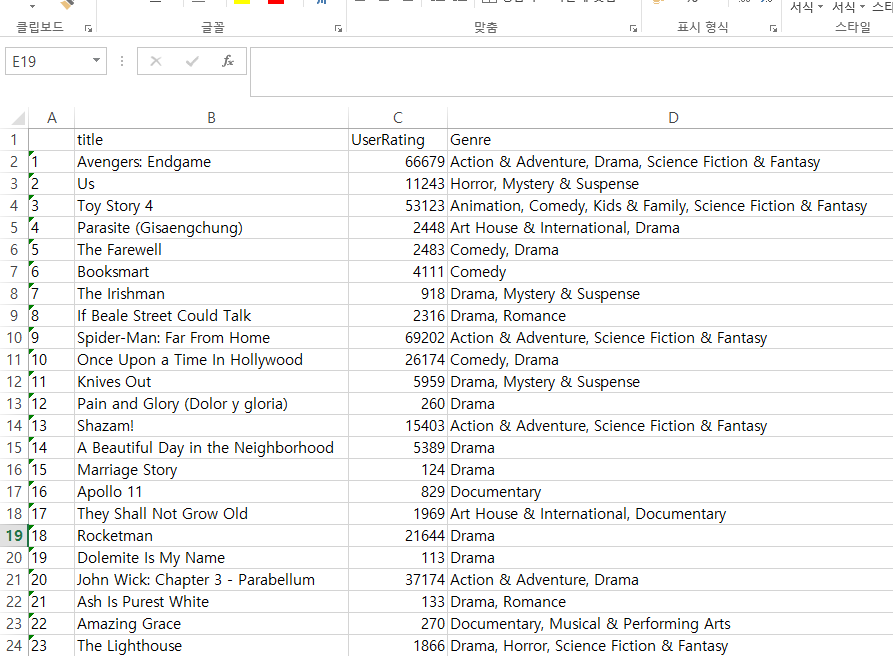
모든 코드가 문제없이 잘 수행되었습니다. 감사합니다!! ☺

**<문제>**

1. **로튼토마토**

로튼 토마토 사이트에서 2019년 가장인기있는 영화 100개에 대해서 영화제목, User Rating, genre, 부분을 추출해서 Data Frame으로 만들고 파일로 출력하라.

**[ Excel 출력결과 ]**



**[ 코드 ]**

|  |
| --- |
| library(rvest)  library(stringr)  library(httr)  # ------url 생성------  urlmain <- "https://www.rottentomatoes.com/top/bestofrt/?year=2019"  html\_page\_w <- read\_html(urlmain);  nodes <- html\_nodes(html\_page\_w,  "td> a.unstyled.articleLink ")  links <- nodes %>% html\_attr("href")  url <- "https://www.rottentomatoes.com"  request\_url <- str\_c(url,links)  #------함수 만들기-----  moviecrawl <-function(rank) {  #-----주소---  html\_page = read\_html(request\_url[rank])    #---제목---  nodes\_ttl <- html\_nodes(html\_page,  "h1")[1]  title<- html\_text(nodes\_ttl, trim = T)  #--user rate---  nodes\_rt <- html\_nodes(html\_page,"strong.mop-ratings-wrap\_\_text--small")[2]  nodes\_rt    rt<-html\_text(nodes\_rt)  rtsep<-str\_extract\_all(html\_text(nodes\_rt),"[0-9]")[1]  rtSepCh <-paste(rtsep[[1]],collapse="")    UserRating <- as.numeric(rtSepCh)    #---Gerne0----  nodes\_gr <- html\_nodes(html\_page,"ul>li:nth-child(2)>div.meta-value>a ")  genresep <-html\_text(nodes\_gr)  Genre <-paste(genresep,collapse=", ")    indMovie <- data.frame(title, UserRating, Genre)    return(indMovie)  }  # ----------결합방법 1 ---------------  movieresult\_df = data.frame();  for(i in 1:length(request\_url)){  movieresult\_df<-rbind(movieresult\_df, moviecrawl(i))  }  View(movieresult\_df)  # -----------결합방법 2---------------------------  movieresult\_df<-vector()  for(i in 1:length(request\_url)){  movieresult\_df<-rbind(movieresult\_df, moviecrawl(i))  }  View(movieresult\_df)  #-----------------저장---------------------  write.csv(movieresult\_df, file = "C:/r\_lecture/data/movieresult.csv")  library(xlsx)  write.xlsx(movieresult\_df, file = "C:/r\_lecture/data/movieresult.xlsx") |

1. **Mpg data set EDA 연습문제**

**[결과]**

|  |
| --- |
| **1. displ(배기량)이 4 이하인 자동차와 5 이상인 자동차 중**  어떤 자동차의 hwy(고속도로 연비)가 평균적으로 더 높은가?    **2. 자동차 제조 회사에 따라 도시 연비가 다른지 알아보려고 한다. "audi"와 "toyota" 중 어느 manufacturer(제조회사)의 cty(도시 연비)가 평균적으로 더 높은지 확인하세요.**    **3. "chevrolet", "ford", "honda" 자동차의 고속도로 연비 평균을 알아보려고 한다. 이 회사들의 데이터를 추출한 후 hwy(고속도로 연비) 전체 평균을 구하세요.**    **4. "audi"에서 생산한 자동차 중에 어떤 자동차 모델의 hwy(고속도로 연비)가 높은지 알아보려고 한다. "audi"에서 생산한 자동차 중 hwy가 1~5위에 해당하는 자동차의 데이터를 출력하세요.**    **# 5. mpg 데이터는 연비를 나타내는 변수가 2개입니다. 두 변수를 각각 활용하는 대신 하나의 통합 연비 변수를 만들어 사용하려 합니다. 평균 연비 변수는 두 연비(고속도로와 도시)의 평균을 이용합니다. 회사별로 "suv" 자동차의 평균 연비를 구한후 내림차순으로 정렬한 후 1~5위까지 데이터를 출력하세요.**    **# 6. mpg 데이터의 class는 "suv", "compact" 등 자동차의 특징에 따라 일곱 종류로 분류한 변수입니다. 어떤 차종의 도시 연비가 높은지 비교하려 합니다. class별 cty 평균을 구하고 cty 평균이 높은 순으로 정렬해 출력하세요.**    **# 7. 어떤 회사 자동차의 hwy(고속도로 연비)가 가장 높은지 알아보려 합니다. hwy(고속도로 연비) 평균이 가장 높은 회사 세 곳을 출력하세요.**    **# 8. 어떤 회사에서 "compact" 차종을 가장 많이 생산하는지 알아보려고 합니다. 각 회사별 "compact" 차종 수를 내림차순으로 정렬해 출력하세요.** |

**[코드]**

|  |
| --- |
| install.packages("ggplot2")  install.packages("dplyr")  library(ggplot2)  library(dplyr)  df = as.data.frame(mpg) # mpg data frame  **# 1. displ(배기량)이 4 이하인 자동차와 5 이상인 자동차 중**  # 어떤 자동차의 hwy(고속도로 연비)가 평균적으로 더 높은가?  df %>% mutate(tmp=ifelse(displ <= 4, "LOWER4",  ifelse(displ >=5,"UPPER5","NONE"))) %>%  group\_by(tmp) %>%  summarise(avg\_hwy = mean(hwy)) %>%  as.data.frame() %>%  filter(tmp != "NONE")  **# 2. 자동차 제조 회사에 따라 도시 연비가 다른지 알아보려고 한다. "audi"와 "toyota" 중 어느 manufacturer(제조회사)의 cty(도시 연비)가 평균적으로 더 높은지 확인하세요.**  df %>% filter(manufacturer == "audi" | manufacturer == "toyota") %>%  group\_by(manufacturer) %>%  summarise(avg\_rate = mean(cty))  **# 3. "chevrolet", "ford", "honda" 자동차의 고속도로 연비 평균을 알아보려고 한다. 이 회사들의 데이터를 추출한 후 hwy(고속도로 연비) 전체 평균을 구하세요.**  df %>% filter(manufacturer == "chevrolet" |  manufacturer == "ford" |  manufacturer == "honda") %>%  summarise(avg\_hwy = mean(hwy))  **# 4. "audi"에서 생산한 자동차 중에 어떤 자동차 모델의 hwy(고속도로 연비)가 높은지 알아보려고 한다. "audi"에서 생산한 자동차 중 hwy가 1~5위에 해당하는 자동차의 데이터를 출력하세요.**  df %>% filter(manufacturer == "audi") %>%  arrange(desc(hwy)) %>%  head(5)  **# 5. mpg 데이터는 연비를 나타내는 변수가 2개입니다. 두 변수를 각각 활용하는 대신 하나의 통합 연비 변수를 만들어 사용하려 합니다. 평균 연비 변수는 두 연비(고속도로와 도시)의 평균을 이용합니다. 회사별로 "suv" 자동차의 평균 연비를 구한후 내림차순으로 정렬한 후 1~5위까지 데이터를 출력하세요.**  df %>% filter(class == "suv") %>%  group\_by(manufacturer) %>%  summarise(avg\_rate = mean(c(cty,hwy))) %>%  arrange(desc(avg\_rate)) %>%  head(5)  **# 6. mpg 데이터의 class는 "suv", "compact" 등 자동차의 특징에 따라 일곱 종류로 분류한 변수입니다. 어떤 차종의 도시 연비가 높은지 비교하려 합니다. class별 cty 평균을 구하고 cty 평균이 높은 순으로 정렬해 출력하세요.**  df %>% group\_by(class) %>%  summarise(avg\_cty = mean(cty)) %>%  arrange(desc(avg\_cty))  **# 7. 어떤 회사 자동차의 hwy(고속도로 연비)가 가장 높은지 알아보려 합니다. hwy(고속도로 연비) 평균이 가장 높은 회사 세 곳을 출력하세요.**  df %>% group\_by(manufacturer) %>%  summarise(avg\_hwy = mean(hwy)) %>%  arrange(desc(avg\_hwy)) %>%  head(3)  **# 8. 어떤 회사에서 "compact" 차종을 가장 많이 생산하는지 알아보려고 합니다. 각 회사별 "compact" 차종 수를 내림차순으로 정렬해 출력하세요.**  df %>% filter(class == "compact") %>%  group\_by(manufacturer) %>%  summarise(n\_compact = n()) %>%  arrange(desc(n\_compact)) |

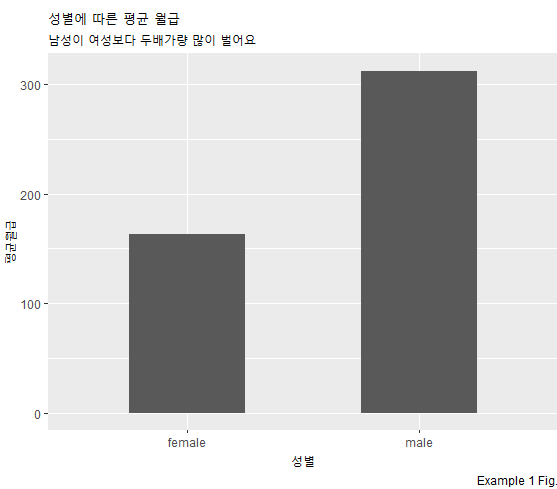
1. **한국복지패널데이터 연습문제**

[데이터불러오기]

|  |
| --- |
| install.packages("foreign")  library(foreign)  library(stringr)  install.packages("foreign")  install.packages("dplyr")  install.packages("ggplot2")  library(foreign)  library(stringr)  library(xlsx)  library(dplyr)  library(ggplot2)  raw\_data\_file <- "C:/r\_lecture/data/한국복지패널데이터/Koweps\_hpc10\_2015\_beta1.sav"  raw\_welfare <- read.spss(raw\_data\_file,  to.data.frame = T)  welfare<-raw\_welfare  head(welfare)  str(welfare)  welfare <- rename(welfare,  gender=h10\_g3, # 성별  birth=h10\_g4, # 출생년도  marriage=h10\_g10, # 결혼 상태  religion=h10\_g11, # 종교 유무  code\_job=h10\_eco9,# 직종 코드  income=p1002\_8aq1,# 월 평균 임금  code\_region=h10\_reg7) |

1. 성별에 따른 월급 차이

**[결과]**

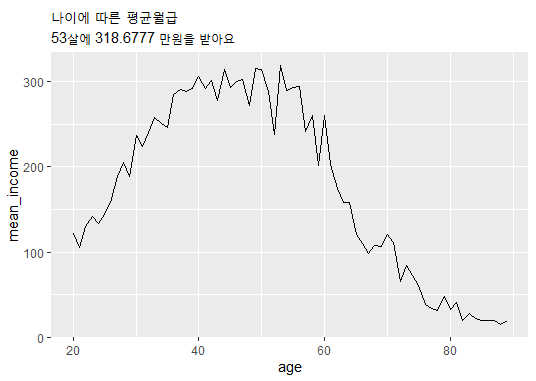


**[코드]**

|  |
| --- |
| table(welfare$gender)  welfare$gender = ifelse(welfare$gender==1,  "male",  "female")  table(welfare$gender)  class(welfare$income)  summary(welfare$income)  qplot(welfare$income)+  xlim(0,1000)  welfare$income = ifelse(welfare$income %in% c(0,9999),  NA,  welfare$income)  table(is.na(welfare$income))  gender\_income <- welfare%>%  filter(!is.na(income))%>% # 결측치 제거  group\_by(gender)%>% # 그룹ing  summarise(mean\_income = mean(income)) # 평균값  gender\_income=as.data.frame(gender\_income)  gender\_income  ggplot(data=gender\_income,  aes(x=gender,  y=mean\_income))+  geom\_col(width = 0.5)+  labs(x="성별",  y="평균월급",  title="성별에 따른 평균 월급",  subtitle="남성이 여성보다 두배가량 많이 벌어요",  caption="Example 1 Fig.") |

1. 나이와 월급의 관계 파악 (몇살에 월급을 가장 많이 받나?, 선그래프표현)

**[결과]**

****

**[코드]**

|  |
| --- |
| summary(welfare$birth)  age\_income<-welfare%>%  mutate(age=2015-birth+1)%>%  filter(income>0&income!=9999&!is.na(income))%>%  group\_by(age)%>%  summarise(mean\_income=mean(income,na.rm = T))%>%  arrange(desc(mean\_income))  ggplot(data=age\_income,  aes(x=age,  y=mean\_income)) +  geom\_line()+  labs(title = "나이에 따른 평균월급",  subtitle="53살에 318.6777 만원을 받아요") |

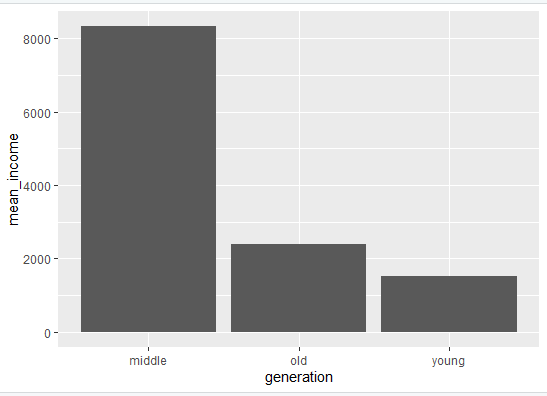
1. 연령대에 따른 월급차이

30대 미만 : 초년(youn)

30~59세: 중년(middle)

60세 이상: 노년(old)

**[결과]**

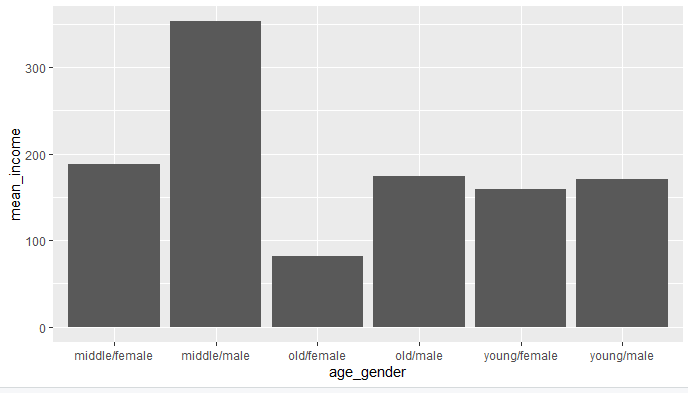
****

**[코드]**

|  |
| --- |
| generationdf<-welfare%>% mutate(age=2015-birth+1) %>% mutate(generation=age)  for(i in 1:length(generationdf$generation)){  if(generationdf$generation[i]>=1&generationdf$generation[i]<30){  generationdf$generation[i]="young"  }else if(generationdf$generation[i]>29&generationdf$generation[i]<60){  generationdf$generation[i]="middle"  }else if(generationdf$generation[i]>59){  generationdf$generation[i]="old"  }  }  table(generationdf$generation)  # 그래프 그리기  generationdf<-generationdf%>%  group\_by(generation)%>%  summarise(mean\_income=mean(income,na.rm = T))  View(generationdf)  ggplot(generationdf,  aes(x=generation,  y=mean\_income))+  geom\_col() |

1. 연령대 및 성별의 월급 차이

**[결과]**

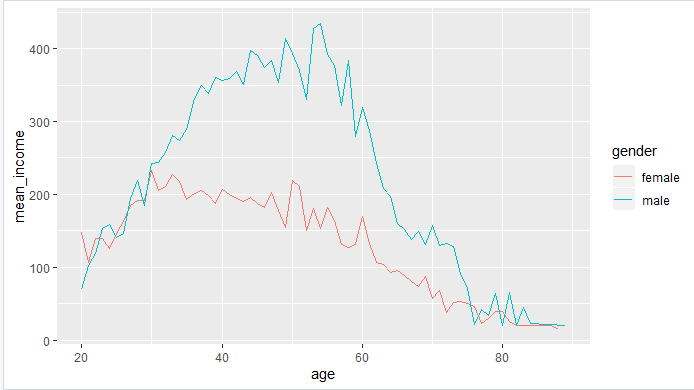
****

**[코드]**

|  |
| --- |
| welfare<-welfare%>%  mutate(age\_group =ifelse(age<30,"young",  ifelse(age<60,  "middle",  "old")))  gender\_age\_income<-welfare%>%  mutate(age\_gender = age\_group)%>%  mutate(age\_gender = ifelse(gender=="female",  str\_c(age\_gender,"/",gender),  str\_c(age\_gender,"/",gender)))%>%  group\_by(age\_gender)%>%  summarise(mean\_income=mean(income,na.rm = T))  ggplot(data=gender\_age\_income,  aes(x=age\_gender,  y=mean\_income))+  geom\_col() |

1. 나이 및 성별에 따른 월급 차이 분석

**[결과]**

****

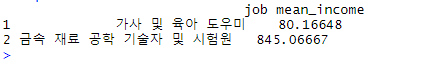
**[코드]**

|  |
| --- |
| gender\_age <- welfare%>%  filter(!is.na(income))%>%  group\_by(age,gender)%>%  summarise(mean\_income=mean(income,na.rm = T))  gender\_age<- as.data.frame(gender\_age)  gender\_age  ggplot(data=gender\_age,  aes(x=age,  y=mean\_income,  color=gender))+  geom\_line() |

1. 직업별 월급 차이를 분석

가장 월급을 많이 받는 직업과, 가장 적게 받는 직업은?

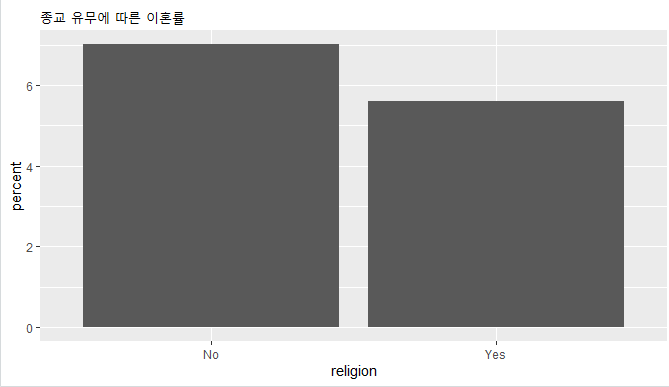
**[결과]**

****

**[코드]**

|  |
| --- |
| install.packages("readxl")  library(readxl)  jobdf <- read\_excel("C:/r\_lecture/data/한국복지패널데이터/Koweps\_Codebook.xlsx",sheet = 2)  jobdf <- as.data.frame(jobdf)  jobdf  exp <- left\_join(welfare,jobdf)  exp$job  jobinc <-  exp %>% filter(!is.na(job))%>% group\_by(job)%>% summarise(mean\_income = mean(income,na.rm = T))  jobinc <- as.data.frame(jobinc)  ## 제일 높은 직업과 낮은직업 뽑기  exp2 <- rbind(head(jobinc %>% arrange(mean\_income),1),head(jobinc %>% arrange(desc(mean\_income)),1))  exp2 |

1. 나이와 월급의 관계 파악

**[결과] **

**[코드]**

|  |
| --- |
| df<- welfare%>%  mutate(divorce=ifelse(marriage %in% c(1,2,3,4),  marriage,NA),  religion=ifelse(religion==1,"Yes","No"))%>%  filter(!is.na(divorce))%>%  group\_by(religion,divorce)%>%  summarise(percent=n())  head(df,8)  df<-df%>%  mutate(total=ifelse(religion=="No",  summarise(group\_by(df,religion),n=sum(percent))[1,2],  summarise(group\_by(df,religion),n=sum(percent))[2,2]))%>%  filter(divorce==3)%>%  mutate(percent=percent/total[[1]]\*100)%>%  select(religion,percent)  df  ggplot(df,  aes(x=religion,  y=percent))+  geom\_col()+  labs(title="종교 유무에 따른 이혼률") |