**[ 박소희 ]**

1. **Rotten Tomatoes**

**# 패키지**

**install.packages("jsonlite")**

**install.packages("rvest")**

**install.packages("httr")**

**library(jsonlite)**

**library(stringr)**

**library(httr)**

**library("rvest")**

**# url**

**origin\_url <-"https://www.rottentomatoes.com/top/bestofrt/?year=2019"**

**main\_url <-"https://www.rottentomatoes.com"**

**# HTML 읽어와서 자료구조화**

**html\_page <- read\_html(origin\_url)**

**html\_node <- html\_nodes(html\_page, "tr>td>a.unstyled.articleLink")**

**# 저장할 곳 만들기**

**title <- vector(mode = "character", length = 100)**

**userrt <- vector(mode = "character", length = 100)**

**genre <- vector(mode = "character", length = 100)**

**# 접속할 링크**

**for(i in 1:100){**

**link <- html\_attr(html\_node,name="href")**

**movie\_link <- str\_c(main\_url,link)[i]**

**html\_info\_page <- read\_html(movie\_link)**

**# 영화제목**

**title\_tmp <- html\_text(html\_nodes(html\_info\_page,**

**"h1"),**

**trim=T)**

**title[i] = title\_tmp**

**# 영화 사용자 점수**

**userrt\_tmp <- html\_text(html\_nodes(html\_info\_page,**

**"strong.mop-ratings-wrap\_\_text--small")[2],**

**trim = TRUE)**

**userrt[i] = userrt\_tmp**

**# 영화 장르**

**genre\_tmp <- html\_text(html\_nodes(html\_info\_page,**

**"div.meta-value>a[href\*=genres]"),**

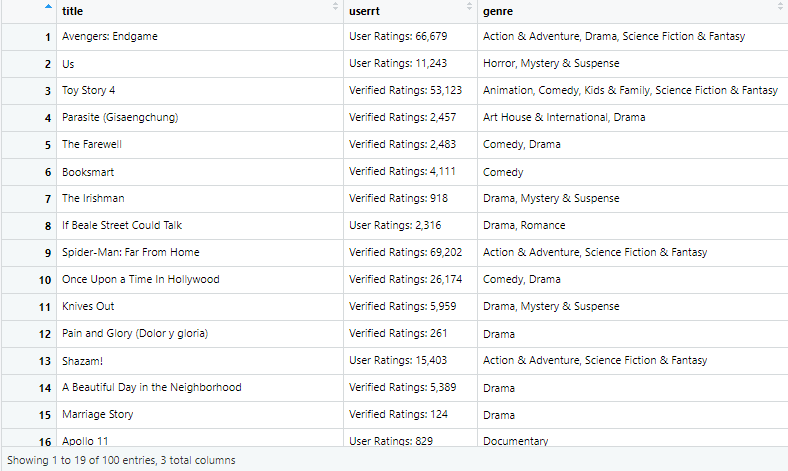
**trim=TRUE)**

**genre[i] = paste(genre\_tmp,collapse=", ")**

**}**

**result<- data.frame(title,userrt,genre)**

**View(result)**



1. **Mpg data set**

**library(ggplot2)**

**library(dplyr)**

**library(xlsx)**

**View(mpg)**

**df <- as.data.frame(mpg)**

**# 1. displ(배기량)이 4 이하인 자동차와 5 이상인 자동차 중 어떤 자동차의 hwy(고속도로 연비)가 평균적으로 더 높은지 확인하세요.**

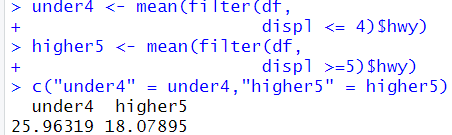
**under4 <- mean(filter(df,**

**displ <= 4)$hwy)**

**higher5 <- mean(filter(df,**

**displ >=5)$hwy)**

**c("under4" = under4,"higher5" = higher5)**

****

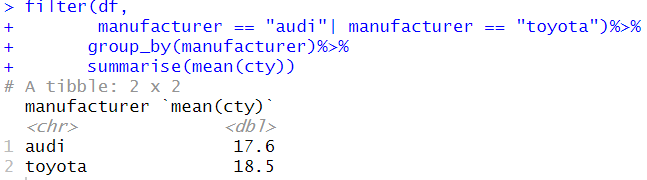
**# 2. 자동차 제조 회사에 따라 도시 연비가 다른지 알아보려고 한다. "audi"와 "toyota" 중 어느 manufacturer(제조회사)의 cty(도시 연비)가 평균적으로 더 높은지 확인하세요.**

**filter(df,**

**manufacturer == "audi"| manufacturer == "toyota")%>%**

**group\_by(manufacturer)%>%**

**summarise(mean(cty))**

****

**# 3. "chevrolet", "ford", "honda" 자동차의 고속도로 연비 평균을 알아보려고 한다. 이 회사들의 데이터를 추출한 후 hwy(고속도로 연비) 전체 평균을 구하세요.**

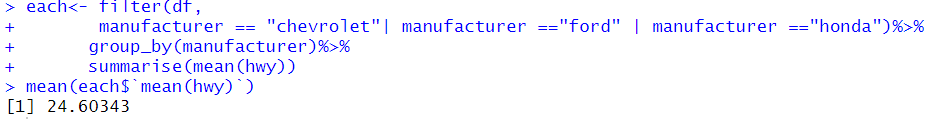
**each<- filter(df,**

**manufacturer == "chevrolet"| manufacturer =="ford" | manufacturer =="honda")%>%**

**group\_by(manufacturer)%>%**

**summarise(mean(hwy))**

**mean(each$`mean(hwy)`)**

****

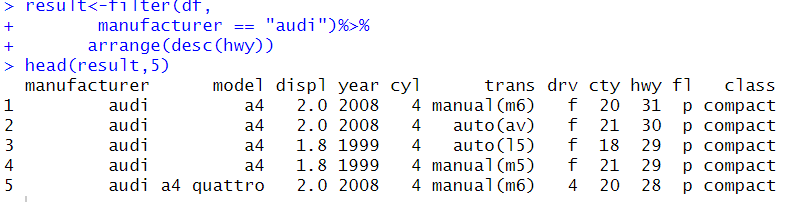
**# 4. "audi"에서 생산한 자동차 중에 어떤 자동차 모델의 hwy(고속도로 연비)가 높은지 알아보려고 한다. "audi"에서 생산한 자동차 중 hwy가 1~5위에 해당하는 자동차의 데이터를 출력하세요.**

**result<-filter(df,**

**manufacturer == "audi")%>%**

**arrange(desc(hwy))**

**head(result,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.**

**new\_df <- mutate(df,**

**average = (cty+hwy)/2)**

**summary <- filter(new\_df,**

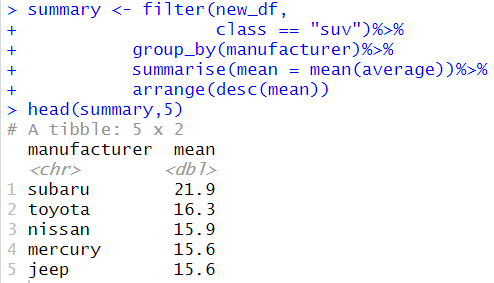
**class == "suv")%>%**

**group\_by(manufacturer)%>%**

**summarise(mean = mean(average))%>%**

**arrange(desc(mean))**

**head(summary,5)**

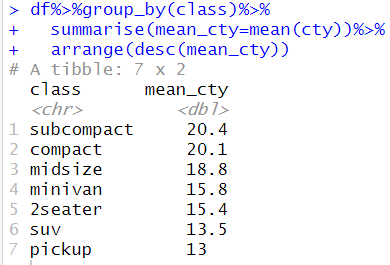
****

**# 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(mean\_cty=mean(cty))%>%**

**arrange(desc(mean\_cty))**

****

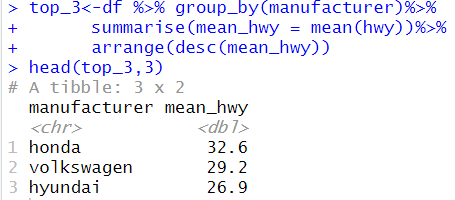
**# 7. 어떤 회사 자동차의 hwy(고속도로 연비)가 가장 높은지 알아보려 합니다. hwy(고속도로 연비) 평균이 가장 높은 회사 세 곳을 출력하세요.**

**top\_3<-df %>% group\_by(manufacturer)%>%**

**summarise(mean\_hwy = mean(hwy))%>%**

**arrange(desc(mean\_hwy))**

**head(top\_3,3)**

****

**# 8. 어떤 회사에서 "compact" 차종을 가장 많이 생산하는지 알아보려고 합니다. 각 회사별 "compact" 차종 수를 내림차순으로 정렬해 출력하세요.**

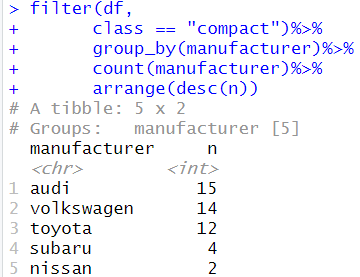
**filter(df,**

**class == "compact")%>%**

**group\_by(manufacturer)%>%**

**count(manufacturer)%>%**

**arrange(desc(n))**

****

1. **한국복지패널데이터**

**@ 1. 성별에 따른 월급 차이**

**# 이상치 확인**

**table(welfare$gender)**

**# 1=남자, 2=여자**

**welfare$gender = ifelse(welfare$gender == 1,**

**"male",**

**"female")**

**# 데이터타입 확인**

**class(welfare$income)**

**# 기본 통계량 확인**

**summary(welfare$income)**

**# 이상치 처리**

**welfare$income = ifelse(welfare$income %in% c(0,9999),**

**NA,**

**welfare$income)**

**# NA개수 확인**

**table(is.na(welfare$income))**

**# 사용할 데이터를 골라내보자**

**gender\_income <- welfare %>%**

**filter(!is.na(income)) %>%**

**group\_by(gender) %>%**

**summarise(mean\_income = mean(income))**

**gender\_income = as.data.frame(gender\_income)**

**gender\_income**

**# 그래프로 그리기**

**ggplot(data=gender\_income,**

**aes(x=gender,**

**y=mean\_income)) +**

**geom\_col(width=0.5) +**

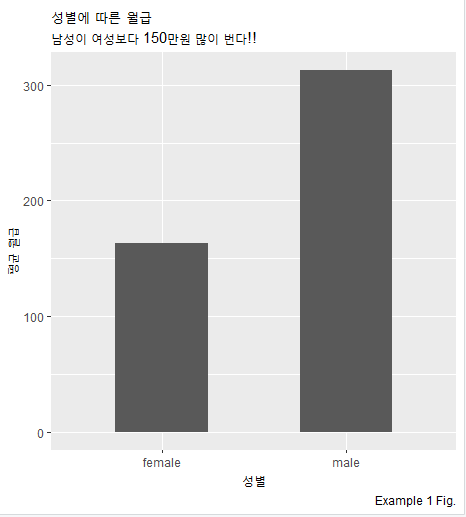
**labs(x="성별",**

**y="평균 월급",**

**title="성별에 따른 월급",**

**subtitle="남성이 여성보다 150만원 많이 번다!!",**

**caption="Example 1 Fig.")**

****

**@ 2. 나이와 월급의 관계 파악해보자, 몇 살 때 월급을 가장 많이 받을까?**

**# 출생연도 파악하기**

**class(welfare$birth)**

**summary(welfare$birth)**

**# 나이 결측치 확인**

**table(is.na(welfare$birth))**

**# 이상치 확인**

**welfare$birth = ifelse(welfare$birth == 9999,**

**NA,**

**welfare$birth)**

**table(is.na(welfare$birth))**

**# 나이 column 생성**

**welfare <- welfare %>%**

**mutate(age = 2015 - birth + 1)**

**# 나이별로 수입평균 계산**

**age\_income <- welfare %>%**

**filter(!is.na(income)) %>%**

**group\_by(age) %>%**

**summarise(mean\_income = mean(income))**

**head(age\_income)**

**age\_income = as.data.frame(age\_income)**

**# 평균 월급에 따라 내림차순했을 때 첫번째 출력**

**age\_income %>% arrange(desc(mean\_income)) %>%**

**select(age) %>% head(1)**

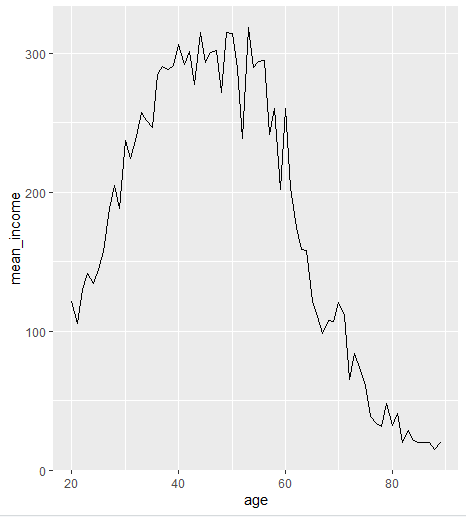
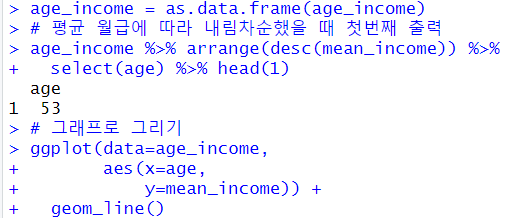
**# 그래프로 그리기**

**ggplot(data=age\_income,**

**aes(x=age,**

**y=mean\_income)) +**

**geom\_line()**

****

**@ 3. 연령대에 따른 월급 차이**

**# 30대 미만 : 초년(young)**

**# 30~59세 : 중년(middle)**

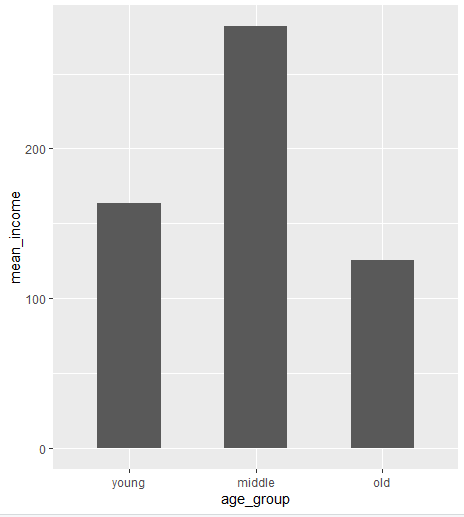
**# 60세 이상 : 노년(old)**

**# 위의 범주로 연령대에 따른 월급 차이 분석**

**# 연령대 column 생성**

**welfare <- welfare %>%**

**mutate(age\_group = ifelse(age < 30,**

** "young",**

**ifelse(age<60,**

**"middle",**

**"old")))**

**# 연령대별 평균 수입 출력**

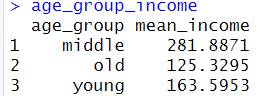
**age\_group\_income <- welfare %>%**

**filter(!is.na(income)) %>%**

**group\_by(age\_group) %>%**

**summarise(mean\_income = mean(income))**

**age\_group\_income = as.data.frame(age\_group\_income)**

**age\_group\_income**

**ggplot(data=age\_group\_income,**

**aes(x=age\_group,**

**y=mean\_income)) +**

**geom\_col(width=0.5) +**

**scale\_x\_discrete(limits=c("young","middle","old"))**

**@ 4. 연령대 및 성별의 월급 차이를 알아보아요**

**# 초년 여자, 초년 남자**

**# 중년 여자, 중년 남자**

**# 노년 여자, 노년 남자**

**# 위에서 나눈 그룹 사용하기**

**gender\_age\_income <- welfare %>%**

**filter(!is.na(income)) %>%**

**group\_by(age\_group, gender) %>%**

**summarise(mean\_income = mean(income))**

**gender\_age\_income = as.data.frame(gender\_age\_income)**

**plot.new()**

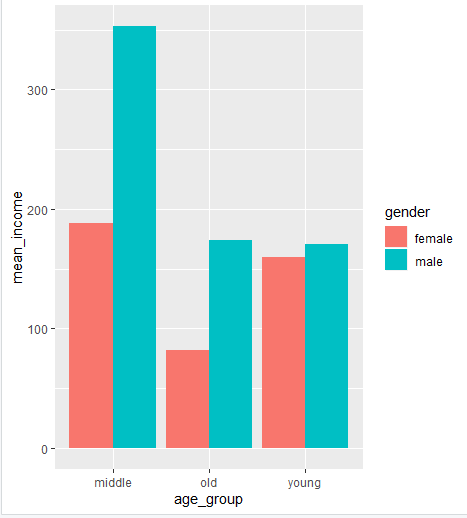
**# 누적차트 만들기**

**ggplot(data=gender\_age\_income,**

**aes(x=age\_group,**

**y=mean\_income,**

**fill=gender)) +**

** geom\_col(position="dodge")**

**@ 5. 나이 및 성별에 따른 월급 차이 분석**

**welfare <- welfare %>%**

**mutate(age = 2015 - birth + 1)**

**# 나이별로 수입평균 계산**

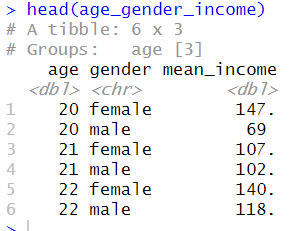
**age\_gender\_income <- welfare %>%**

**filter(!is.na(income)) %>%**

**group\_by(age, gender) %>%**

**summarise(mean\_income = mean(income))**

**head(age\_gender\_income)**

****

**@ 6. 직업별 월급**

**welfare <- welfare %>%**

**mutate(job\_group)**

**# 직업별 평균 수입 출력**

**job\_group\_income <- welfare %>%**

**filter(!is.na(income)) %>%**

**group\_by(code\_job) %>%**

**summarise(mean\_income = mean(income))**

**job\_group\_income = as.data.frame(job\_group\_income)**

**job\_group\_income**

**# 코드북 불러오기**

**code\_book <- read.xlsx(file.choose(),sheetIndex=2,encoding = "UTF-8")**

**result<-left\_join(job\_group\_income,code\_book,by="code\_job")**

**View(result)**

****

**@ 7.** **종교 유무에 따른 이혼률**

**# 종교가 없는 사람의 이혼율 : 7.0**

**# 종교가 있는 사람의 이혼율 : 5.6**

**# 1,2,4면 group\_marriage => marriage**

**# 3이면 group\_marriage => divorce**

**# 이혼여부에 관한 column 생성**

**welfare <-welfare %>%**

**mutate(group\_marriage**

**= ifelse(marriage%in%c(1,2,4),**

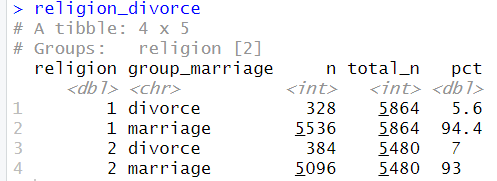
**"marriage",**

**ifelse(marriage == 3,**

**"divorce",**

**NA)))**

**table(welfare$group\_marriage)**

****

**# 종교유무에 따른 이혼여부 출력**

**religion\_divorce <- welfare %>%**

**filter(!is.na(group\_marriage))%>%**

**group\_by(religion, group\_marriage)%>%**

**summarise(n=n())%>%**

**mutate(total\_n = sum(n))%>%**

**mutate(pct = round(n/total\_n\*100,1))**

**religion\_divorce**