```
2week 변수와 벡터
log(10)+5 # 로그함수 / sqrt(25) # 제곱근 /max(5,3,2) # 가장 큰 값
a <- "A" # a에 문자 저장 a+b # 에러 발생
v1 <- 50:90 # 50~90의 연속적인 벡터 생성 v2 <- c(1,2,5, 50:90) #1,2,5 출력되고 같음
v3 <- seq(1,101,3) # 1~101까지 3간격으로 저장, 101이 넘어가면 저장X / v4 <- seq(0.1,1.0,0.1)
# 0.1~1지 0.1간격으로 저장
v6 <- rep(1:5,times=3) # 1에서 5까지 3번 반복 / v7 <- rep(c(1,5,9), times=3) # 1, 5, 9를 3번
반복,"rep(1,5,9, times=3)"와는 결과 다름
# a)3 벡터의 이해================================================
score <- c(90,85,70) # 성적
names(score) # score에 저장된 값들의 이름을 보이시오
names(score) <- c("John","Tom","Jane") # 값들에 이름을 부여
names(score) # score에 저장된 값들의 이름을 보이시오
score # 이름과 함께 값이 출력
d <- c(1,4,3,7,8)
d[c(1,3,5)] # 1, 3, 5인덱스의 값 출력
d[1:3] # 1~3 인덱스의 값 출력
d[seq(1,5,2)] # 홀수 번째 값 출력
d[-2] # 2번째 값 제외하고 출력
d[-c(3:5)] # 3~5번째 값은 제외하고 출력
# a)4 벡터의 연산================================================
d <- c(1,2,3,4,5,6,7,8,9,10)
sum(d) # d의 포함된 값들의 합
sum(2*d) # d의 포함된 값들에 2를 곱한 후 합한 값
length(d) # d에 포함된 값들의 개수
mean(d[1:5]) # 1~5번째 값들의 평균
max(d) # d에 포함된 값들의 최댓값
min(d) # d에 포함된 값들의 최솟값
sort(d) # 오름차순 정렬
sort(d, decreasing = FALSE) # 오름차순 정렬
sort(d, decreasing = TRUE) # 내림차순 정렬
# a) 리스트, 팩터================================================
ds <- c(90, 85, 70, 84)
my.info <- list(name='Tom', age=60, status=TRUE, score=ds) # 하나의 튜플
my.info # 리스트에 저장된 내용을 모두 출력
my.info[[1]] # 리스트의 첫 번째 값을 출력
my.info$name # 리스트에서 값의 이름이 name인 값을 출력
my.info[[4]] # 리스트의 네 번째 값을 출력
bt <- c('A', 'B', 'B', 'O', 'AB', 'A') # 문자형 벡터 bt 정의
bt.new <- factor(bt) # 팩터 bt.new 정의
bt # 벡터 bt의 내용 출력
bt.new # 팩터 bt.new의 내용 출력
bt[5] # 벡터 bt의 5번째 값 출력
bt.new[5] # 팩터 bt.new의 5번째 값 출력
levels(bt.new) # 팩터에 저장된 값의 종류를 출력
as.integer(bt.new) # 팩터의 문자값을 숫자로 바꾸어 출력
bt.new[7] <- 'B' # 팩터 bt.new의 7번째에 'B' 저장
bt.new[8] <- 'C' # 팩터 bt.new의 8번째에 'C' 저장 X
bt.new # 팩터 bt.new의 내용 출력
3week (메트릭스와 데이터 프레임)================================
# 메트리스 생성 (기본 열부터 채움, byrow=T - 행부터 채움) 데이터프레임은 여려 개의 벡터를
세로 방향으로 묶어 놓은 개념
z <- matrix(1:20, nrow=4, ncol=5)# 데이터의 개수와 행*열 개수가 달라도 가능
m1 <- cbind(x,y) # x와 y를 열 방향으로 결합하여 매트릭스 생성
m2 <- rbind(x,y) # x와 y를 행 방향으로 결합하여 매트릭스 생성
m3 <- rbind(m2,x) # 매트릭스 m2와 벡터 x를 행 방향으로 결합
z[1,c(1,2,4)] # 1행의 값 중 1, 2, 4열에 있는 값
rownames(score) <- c('John','Tom','Mark','Jane') # 행 이름 지정
colnames(score) <- c('English','Math','Science') # 열 이름 지정
# 데이터 프레임================================================
city <- c("Seoul","Tokyo","Washington") # 문자로 이루어진 벡터
rank <- c(1,3,2) # 숫자로 이루어진 벡터
city.info <- data.frame(city, rank) # 데이터프레임 생성
iris[,c("Sepal.Length","Species")] # 1, 5열의 모든 데이터
dim(iris) # 행과 열의 개수 출력
nrow(iris) # 행의 개수 출력
ncol(iris) # 열의 개수 출력
colnames(iris) # 열 이름 출력, names( )와 결과 동일
head(iris) # 데이터셋의 앞부분 일부 출력
tail(iris) # 데이터셋의 뒷부분 일부 출력
str(iris) # 데이터셋 요약 정보 보기
iris[,5] # 품종 데이터 보기
unique(iris[,5]) # 품종의 종류 보기(중복 제거)
table(iris["Species"]) # 품종의 종류별 행의 개수 세기
colSums(iris[,-5]) # 열별 합계 # 아이리스 5번째 품종 범주형이여서 -5 제외함
colMeans(iris[,-5]) # 열별 평균
rowSums(iris[,-5]) # 행별 합계
rowMeans(iris[,-5]) # 행별 평균
t(z) # 행과열 방향 전환
# 조건에 맞는 행과 열 출력
IR.1 <- subset(iris, Species=="setosa")
IR.2 <- subset(iris, Sepal.Length>5.0 &
Sepal.Width>4.0)
class(iris) # iris 데이터셋의 자료구조 확인
is.matrix(iris) # 데이터셋이 매트릭스인지를 확인하는 함수
iris["Species"] # 결과=벡터. 매트릭스와 데이터프레임 모두 가능 열추출
iris[,5] / iris["Species"] / iris[5] / iris$Species # 이건 해당 해당 $ 다 출력
================================================================
setwd("C:/r_workspace") # 작업 폴더 지정
air <- read.csv("airquality.csv", header=T) # .csv 파일 읽기
my.iris <- subset(iris, Species='Setosa') # Setosa 품종 데이터만 추출
write.csv(my.iris, "my_iris.csv", row.names=F) # .csv 파일에 저장하기
4week 단일변수 자료 탐색 ========================================
# 범주형 단일 변수 처리
favorite <- c('WINTER', 'SUMMER', 'SPRING', 'SUMMER', 'SUMMER',
'FALL', 'FALL', 'SUMMER', 'SPRING', 'SPRING')
favorite # favorite의 내용 출력
table(favorite) # 도수분포표 계산
table(favorite)/length(favorite) # 비율 출력
> table(favorite) # 도수분포표 계산
favorite
  FALL SPRING SUMMER WINTER
     2      3      4      1
> table(favorite)/length(favorite) # 비율 출력
favorite
  FALL SPRING SUMMER WINTER
   0.2    0.3    0.4    0.1
ds <- table(favorite)
barplot(ds, main='favorite season') # 막대 그래프 / pie()로 바꾸면 원그래프
favorite.color <- c(2, 3, 2, 1, 1, 2, 2, 1, 3, 2, 1, 3, 2, 1, 2)
colors <- c('green', 'red', 'blue')
names(ds) <- colors #자료값 1,2,3을 green, red, blue로 변경
barplot(ds, main='favorite color', col=colors) # 색 지정 막대그래프
pie(ds, main='favorite color', col=colors) # 색 지정 원그래프
================================================================
# 연속형 단일 변수 처리
weight <- c(60, 62, 64, 65, 68, 69)
weight.heavy <- c(weight, 120) # weight 벡터에 120 값 추가
mean(weight) # 평균
median(weight) # 중앙값 (가운데 값 문자 그대로)
mean(weight, trim=0.2) # 절사평균(상하위 20% 제외한 평균 계산 )
# 4분위수 quantile (Q1, Q2, Q3으로 나눔 (4개의 구간 25%))
> mydata <- c(60, 62, 64, 65, 68, 69, 120)
> quantile(mydata)
   0%   25%   50%   75%  100%
 60.0  63.0  65.0  68.5 120.0
> quantile(mydata, (0:10)/10) # 10% 단위로 구간을 나누어 계산
   0%   10%   20%   30%   40%   50%   60%   70%   80%   90%  100%
 60.0  61.2  62.4  63.6  64.4  65.0  66.8  68.2  68.8  89.4 120.0
> summary(mydata)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   60.00   63.00   65.00   72.57   68.50  120.00
# 산포 ========================================================
var(mydata) # 분산 | sd(mydata) # 표준편차
range(mydata) # 값의 범위 | diff(range(mydata)) # 최댓값, 최솟값의 차이
# 히스토그램 (막대 그래프와 달리 간격 x )=============================
dist <- cars[,2] # 자동차 제동거리
hist(dist, # 자료(data)
     main="Histogram for 제동거리", # 제목
     xlab ="제동거리", # x축 레이블
     ylab="빈도수", # y축 레이블
     border="blue", # 막대 테두리색
     col="green", # 막대 색
     las=2, # x축 글씨 방향(0~3)
     breaks=5) # 막대 개수 조절
# 상자그림 ====================================================
dist <- cars[,2]  # 자동차 제동거리(단위: 피트) / 기본 데이터 cars
boxplot(dist, main="자동차 제동거리")  # 기본 상자 그리기
boxplot.stats(dist)
boxplot(Petal.Length~Species, data=iris, main="품종별 꽃잎의 길이") # 세로축 먼저 채움
# 한번에 여러 상자 출력
par(mfrow=c(1,3))  # 가상화면 분할
barplot(table(mtcars$carb),
        main="Barplot of Carburetors",
        xlab="#of carburetors",
        ylab="frequency",
        col="blue")
barplot(table(mtcars$cyl),
        main="Barplot of Cylender",
        xlab="#of cylender",
        ylab="frequency",
        col="red")
barplot(table(mtcars$gear),
        main="Barplot of Grar",
        xlab="#of gears",
        ylab="frequency",
        col="green")
par(mfrow=c(1,1))  # 분할 해제
# 조건에 맞는 위치 찾기=========================================
which(score==69) # 성적이 69인 학생은 몇 번째에 있나
which.max(score) # 최고 점수는 몇 번째에 있나
score <- c(76, 84, 69, 50, 95, 60, 82, 71, 88, 84)
idx <- which(score<=60) # 성적이 60 이하인 값들의 인덱스
score[idx] <- 61          # 성적이 60 이하인 값들은 61점으로 성적 상향 조정
idx <- which(score>=80) # 성적이 80 이상인 값들의 인덱스
score.high <- score[idx] # 성적이 80 이상인 값들만 추출하여 저장
idx <- which(iris$Petal.Length>5.0) # 꽃잎의 길이가 5.0 이상인 값들의 인덱스
iris.big <- iris[idx,]          # 인덱스에 해당하는 값만 추출하여 저장
idx <- which(iris[,1:4]>5.0, arr.ind =TRUE) # 1~4열의 값 중 5보다 큰 값의 행과 열의 위치 # ar
r.ind =TRUE 안하면 디폴트는 F이라서 그냥 해당 원소만 출력됨
> which(m %% 3 == 0)
[1]  3  6  9 12
> which(m %% 3 == 0, arr.ind = TRUE)
     row col
[1,]   3   1
[2,]   3   2
[3,]   3   3
[4,]   3   4
5week 다중변수 자료 탐색 ========================================
# 산점도
wt <-mtcars$wt # 중량 자료  (이건 데이터 메트릭스)
mpg <- mtcars$mpg # 연비 자료
plot(wt, mpg, # 2개 변수(x축, y축)
     main="중량-연비 그래프", # 제목
     xlab="중량", # x축 레이블
     ylab="연비(MPG)", # y축 레이블
     col="red", # point의 color
     pch=11) # point의 종류
# 다중 변수 산점도
vars <- c("mpg","disp","drat","wt") # 대상 변수
target <- mtcars[,vars]
head(target)
pairs(target, # 대상 데이터 # 여기가 다름
      main="Multi Plots")
# 만약 그룹 정보(범주형)의 정보가 있는 두 변수를 산점도로 표현한다면 ?
iris.2 <- iris[,3:4] # 데이터 준비
point <- as.numeric(iris$Species) # 점의 모양 ( numeric을 사용하여 품종 숫자열로 )
point # point 내용 출력
color <- c("red","green","blue") # 점의 컬러
plot(iris.2,
     main="Iris plot",  # 제목 이름
     pch=c(point),   # 점 모양
     col=color[point]) # 컬러 포인터 끌고오는 기법
================================================================
# 상관 분석 ( 선형적 관계 / 피어슨 상관계수 (-1<_r<_1), 양이면 비례, 1 근접시-상관높
beers = c(5,2,9,8,3,7,3,5,3,5) # 자료 입력
bal <- c(0.1,0.03,0.19,0.12,0.04,0.0095,0.07,0.06,0.02,0.05) #자료 입력
tbl <- data.frame(beers,bal) # 데이터프레임생성
plot(bal~beers,data=tbl) # 산점도
res <- lm(bal~beers,data=tbl) # 회귀식 도출
abline(res) # 회귀선 그리기
cor(beers,bal) # 상관계수 계산 / cor(iris[,1:4]) # 4개 변수 간 상관성 분석 (목적차이)
# 선그래프=====================================================
month = 1:12 # 자료 입력   (이건 데이터 벡터)
late = c(5,8,7,9,4,6,12,13,8,6,6,4) # 자료 입력
plot(month, # x data
     late, # y data
     main="지각생 통계", # 제목   기억
     type= "l", # 그래프의 종류 선택(알파벳) 기억
     lty=1, # 선의 종류(line type) 선택  기억
     lwd=1, # 선의 굵기 선택
     xlab="Month", # x축 레이블
     ylab="Late cnt" # y축 레이블
)
# 복수의 선 그래프 (진짜 그래프에 물리적으로 복수해주고 싶다 겁나 많네;;;)
lines(month, # x data
      late2, # y data
      type = "b", # 선의 종류(line type) 선택
      col = "blue") # 선의 색 선택          /  ylim=c(1, 15) # y축 값의 (하한, 상한)
# 자료 탐색
grp <- c()
for (i in 1:nrow(myds)) { # myds$medv 값에 따라 그룹 분류
  if (myds$medv[i] >= 25.0) {
    grp[i] <- "H"          # 25.0 이상이면 상(H), 17.0 이하이면 하(L), 나머지를 중(M)
  } else if (myds$medv[i] <= 17.0) {
    grp[i] <- "L"
  } else {
    grp[i] <- "M"
  }
}
grp <- factor(grp) # 문자벡터를 팩터 타입으로 변경
grp <- factor(grp, levels=c("H","M","L")) # 레벨의 순서를 H,L,M -> H,M,L
myds <- data.frame(myds, grp) # myds 에 grp 컬럼추가
# 데이터 파악 ================================================
str(myds)    # 객체 구조 간략하게 설명
head(myds)
table(myds$grp) # 주택 가격 그룹별 분포
#다중 히스토그램 출력해서 분석
par(mfrow=c(2,3)) # 2x3 가상화면 분할
for(i in 1:5) {
  hist(myds[,i], main=colnames(myds)[i], col="yellow")  # 이거 봐라
}
```

```r
#(5) 다중 boxplot 출력
par(mfrow=c(2,3)) # 2x3 가상화면 분할
for(i in 1:5) {
  boxplot(myds[,i], main=colnames(myds)[i])
}
# myds 데이터셋의 특정 변수(crim, rm, dis, tax)를 grp 변수에 따라 그룹화해서 상자
boxplot(myds$crim~myds$grp, main="1인당 범죄율")
boxplot(myds$rm~myds$grp, main="방의 수")
boxplot(myds$dis~myds$grp, main="직업센터까지의 거리")
boxplot(myds$tax~myds$grp, main="재산세")
#myds 데이터셋의 6번째 열을 제외한 모든 열 간의 산점도 행렬을 생성
pairs(myds[ ,-6])
## (8) scatter plot with group (7)에서 컬러 놀이)
point <- as.integer(myds$grp) # 점의 모양 지정
color <- c("red","green","blue") # 점의 색 지정
pairs(myds[,-6], pch=point, col=color[point])
# 변수 간 상관계수 확인
cor(myds[,-6])
6week 데이터 시각화 기법 (기본 시각화, ggplot, 차원축소)
# 트리맵==================================================
treemap(GNI2014,
        index=c("continent","iso3"),#계층구조 설정(대륙-국가)혹은, 타일에 주 이름 표기
        vSize="population", # 타일의 크기
        vColor="GNI", # 타일의 컬러
        type="value", # 타일 컬러링 방법
        title="World's GNI") # 트리맵 제목
# 버블 차트
st <- data.frame(state.x77) # 매트릭스를 데이터프레임으로 변환
symbols(st$Illiteracy, st$Murder, # 원의 x, y 좌표의 열
        circles=st$Population, # 원의 반지름의 열
        inches=0.3, # 원의 크기 조절값
        fg="white", # 원의 테두리 색
        bg="lightgray", # 원의 바탕색
        lwd=1.5, # 원의 테두리선 두께
        xlab="rate of Illiteracy",
        ylab="crime(murder) rate",
        main="Illiteracy and Crime")
text(st$Illiteracy, st$Murder, # 텍스트가 출력될 x, y 좌표
     rownames(st), # 출력할 텍스트
     cex=0.6, # 폰트 크기
     col="brown") # 폰트 컬러
# 모자이크 플롯
head(mtcars)
mosaicplot(~gear+vs, data = mtcars, color=TRUE,
           main ="Gear and Vs") # ~ 독립변수 종속변수 구별 (여기선x 앞에 씀)
# ggplot==================================================
ggplot(df, aes(x=month,y=rain)) + # 그래프를 그릴 데이터 지정 (기억)
  geom_bar(stat="identity", # 막대 높이는 y축에 해당하는 열의 값
           width=0.7, # 막대의 폭 지정 (기억)
           fill="steelblue") + # 막대의 색 지정 (기억)
  ggtitle("월별 강수량") + # 그래프의 제목 지정
  theme(plot.title = element_text(size=25, face="bold", colour="steelblue")) +
  labs(x="월",y="강수량") + # 그래프의 x, y축 레이블 지정
  coord_flip( ) # 그래프를 가로 방향으로 출력 (이름 기억)
# ggplot 히스토그램
ggplot(iris, aes(x=Sepal.Width, fill=Species, color=Species)) +  # binwith 바 넓이, col 막대 윤곽선, fill 막대 내부 색 # x= 작성대상 열1
geom_histogram(binwidth = 0.5, position="dodge") + # dodge 막대들이 겹치지 않고
theme(legend.position="top")  # 디자인
# 산점도
library(ggplot2)
ggplot(data=iris, aes(x=Petal.Length, y=Petal.Width,
                      color=Species)) +
  geom_point(size=3) +
  ggtitle("꽃잎의 길이와 폭") + # 그래프의 제목 지정
  theme(plot.title = element_text(size=25, face="bold", colour="steelblue"))
# 상자그림
library(ggplot2)
ggplot(data=iris, aes(y=Petal.Length, fill=Species)) +
  geom_boxplot( ) # fill내부색 채우고 다른 종까지 표시
# 선그래프
library(ggplot2)
year <- 1937:1960
cnt <- as.vector(airmiles)
df <- data.frame(year,cnt) # 데이터 준비
head(df)
ggplot(data=df, aes(x=year,y=cnt)) + # 선그래프 작성
  geom_line(col="red")
# 차원축소 ================================================
library(Rtsne) / library(ggplot2)
ds <- iris[,-5] # 품종 정보 제외
## 중복 데이터 제거
dup = which(duplicated(ds))
dup # 143번째 행 중복
ds <- ds[-dup,]
ds.y <- iris$Species[-dup] # 중복을 제외한 품종 정보
## t-SNE 실행
tsne <- Rtsne(ds,dims=2, perplexity=10)
## 축소결과 시각화
df.tsne <- data.frame(tsne$Y)
head(df.tsne)
ggplot(df.tsne, aes(x=X1, y=X2, color=ds.y)) +
  geom_point(size=2)
install.packages(c("rgl", "car"))
library("car")
library("rgl")
library("mgcv")
tsne <- Rtsne(ds,dims=3, perplexity=10)
df.tsne <- data.frame(tsne$Y)
head(df.tsne)
# 회귀면이 포함된 3차원 산점도
scatter3d(x=df.tsne$X1, y=df.tsne$X2, z=df.tsne$X3)
# 회귀면이 없는 3차원 산점도
points <- as.integer(ds.y)
color <- c('red','green','blue')
scatter3d(x=df.tsne$X1, y=df.tsne$X2, z=df.tsne$X3,
          point.col = color[points], # 점의 색을 품종별로 다르게
          surface=FALSE) # 회귀면을 표시하지 않음
7week 데이터 전처리 (결측, 특이값 | 데이터 정렬, 분리, 선택 | 샘플링, 조합 / 집계병합
# 결측값 (0으로 치환하던가, 제외하고 계산하던가, 행 불러서 제거하던가 ㅋ)
z <- c(1,2,3,NA,5,NA,8) # 결측값이 포함된 벡터 z
sum(z) # 정상 계산이 안 됨
is.na(z) # NA 여부 확인
sum(is.na(z)) # NA의 개수 확인
sum(z, na.rm=TRUE) # NA를 제외하고 합계를 계산
mean(z, na.rm=TRUE) # NA를 제외하고 평균을 계산
z1 <- c(1,2,3,NA,5,NA,8) # 결측값이 포함된 벡터 z1
z1[is.na(z1)] <- 0 # NA를 0으로 치환
z3 <- as.vector(na.omit(z2)) # NA를 제거하고 새로운 벡터 생성
x[1,2]<- NA; x[1,3]<- NA # NA를 포함하는 test 데이터 생성
x[!complete.cases(x),] # NA가 포함된 행들 출력
y <- x[complete.cases(x),] # NA가 포함된 행들 제거
#특이값 (Na로 대체하고 처리) ============================================
boxplot.stats(st$Income)$out # 특이값 확인
out.val <- boxplot.stats(st$Income)$out # 특이값 추출
st$Income[st$Income %in% out.val] <- NA # 특이값을 NA로 대체
newdata <- st[complete.cases(st),] # NA가 포함된 행 제거
#데이터 정렬==============================================
order(v1)
v1 <- sort(v1) # 오름차순
v2 <- sort(v1, decreasing=T) # 내림차순
order(iris$Sepal.Length)
iris[order(iris$Sepal.Length),] # 오름차순으로 정렬
iris[order(iris$Sepal.Length, decreasing=T),] # 내림차순으로 정렬
iris.new <- iris[order(iris$Sepal.Length),] # 정렬된 데이터를 저장
iris[order(iris$Species, -iris$Petal.Length, decreasing=T),] # 정렬 기준이 2개(- 내림차순 기준)
# 데이터 분리와 선택 ======================================================
sp <- split(iris, iris$Species) # 품종별로 데이터 분리
sp # 분리 결과 확인
summary(sp) # 분리 결과 요약
sp$setosa # setosa 품종의 데이터 확인 (분리 했으니까 확인이 가능한거)
subset(iris, Species == "setosa") # 서브셋으로 데이터 선택
subset(iris, Sepal.Length > 7.5)
subset(iris, Sepal.Length > 5.1 &
         Sepal.Width > 3.9)
subset(iris, Sepal.Length > 7.6,
       select=c(Petal.Length,Petal.Width))
# 데이터 샘플과 조합 ======================================================
x <- 1:100
y <- sample(x, size=10, replace = FALSE) # 비복원추출 ( 한뽑 다시 노 뽑 )
idx <- sample(1:nrow(iris), size=50, replace = FALSE)
iris.50 <- iris[idx,] # 50개의 행 추출
dim(iris.50) # 행과 열의 개수 확인
set.seed(100) # 요 시드 사용하면 샘플할 때마다 같은거 뽑음
sample(1:20, size=5)
combn(1:5,3) # 1~5에서 3개를 뽑는 조합  (총30)
x = c("red","green","blue","black","white")
com <- combn(x,2) # x의 원소를 2개씩 뽑는 조합  (총20)
com
for(i in 1:ncol(com)) { # 조합을 출력
  cat(com[,i], "\n")
}
▶ red green
▶ red blue
# 데이터 집계와 병합 ======================================================
agg <- aggregate(iris[,-5], by=list(iris$Species), FUN=mean) # 평균을 집계 | by 그룹 기준
agg <- aggregate(iris[,-5], by=list(표준편차=iris$Species), FUN=sd) # 표준편차 ( 그룹 다중 o)
agg <- aggregate(mtcars, by=list(cyl=mtcars$cyl, vs=mtcars$vs),FUN=max) # 각 변수 최대값
> z <- merge(x,y, by=c("name")) # 병합 (공통 열이 같은것만
  name math korean
1    a   90     75
2    b   80     60
merge(x,y, all.x=T) # 첫 번째 데이터셋의 행들은 모두 표시되도록 표시안되는건 NA로 출력됨
merge(x,y, all.y=T) # 두 번째 데이터셋의 행들은 모두 표시되도록
merge(x,y, all=T) # 두 데이터셋의 모든 행들이 표시되도록
x <- data.frame(name=c("a","b","c"), math=c(90,80,40))
y <- data.frame(sname=c("a","b","d"), korean=c(75,60,90))
merge(x,y, by.x=c("name"), by.y=c("sname"))    # 열 이름 다를 때 ^^

=======================집일=(Home work)=========================
table(data[, "Gender"]) # 1. 여자의 수와 남자의 수를 각각 구하는 코드
a = sum(data$Smoke == "yes") # 2.흡연자 수
a / nrow(data)  # 2. 전체 데이터 행 갯수를 나눠서 흡연자의 비율 구함
t3 <- table(data$Gender, data$Smoke) # 3.성별에 따른 흡연자
t3_1 <- t3[, "no"] / rowSums(t3) # 3. 각 성별의 비흡연자 비율을 계산
BMI <- data$Weight / (data$Height / 100)^2 # 4. BMI 계산
data <- cbind(data, BMI) # cbind를 사용하여 데이터 프레임에 BMI 열 추가
ba_female <- mean(data$BMI[data$Gender == 'F'])
bs_male <- sd(data$BMI[data$Gender == 'M']) # 5. 성별에 따른 BMI의 표준편차, 평균 계산
# 성별에 따른 BMI 상자그림 (Boxplot을 사용함_세로축부터 기재)
boxplot(BMI ~ Gender, data = data,xlab = "Gender", ylab = "BMI")
mean_smoker <-aggregate(data$BMI, by=list(평균=data$Smoker), FUN = mean) # 흡연 여부에 따른 BMI 평균 agg_gen <- aggregate(data$BMI, by=list(표준편차=data$Gender), FUN = sd)
# 이건 성별에 따른 BMI 표준편차 계산
# 8. 성별에 따른 키와 몸무게의 산점도로 표현하는 코드
wh <- data[, c( 9, 6)]
wh
data$Gender <- factor(data$Gender)
point <- as.numeric(data$Gender) #숫자로 바꿔주는 부분
point # point 내용 출력
data
color <- c("red","blue") # 점의 컬러
plot(wh,
     xlab= "wegiht",
     ylab= "Height",
     pch=c(point),
     col=color[point])
# 11번 문제
# Arc Name 별로 색상 지정
# ggplot2 패키지 소환
library(ggplot2)
# 누락된 값이 있는 행을 제거
data2 <- na.omit(data2)
# Arc Name 별로 색상을 지정하기 위한 벡터를 생성합니다.
colors <- rainbow(length(unique(data2$Arc.Name)))
names(colors) <- unique(data2$Arc.Name)
# ggplot을 사용하여 산점도를 그립니다.
ggplot(data2, aes(x=Year, y=Average.Rating, color=Arc.Name)) +
  geom_point() +
  scale_color_manual(values = colors) +  # 벡터에 정의된 색상을 사용
  theme_minimal() +
  labs(title="Average Rating by Year and Arc Name", x="Year", y="Average Rating")

# 12번 문제
# "Year" 열에 NA가 있는지 확인
unique(data2$Year)
# NA를 포함한 행 제거
data2 <- na.omit(data2)
#"Year" 열을 숫자형으로 변환
data2$Year <- as.numeric(data2$Year)
# Total.Vote 열에서 수치형으로 변환할 수 없는 값들을 확인
non_numeric <- which(!grepl("^[0-9]+$", data2$Total.Vote))
# 수치형으로 변환할 수 없는 값들을 NA로 대체
data2$Total.Vote[non_numeric] <- NA
#"Total.Vote" 열을 숫자형으로 변환
data2$Total.Vote <- as.numeric(data2$Total.Vote)
# 연속형 데이터 따로 저장
myds <- data2[, c("Year", "Total.Vote", "Average.Rating")]
# 모든 열 박스 상자로 출력
par(mfrow=c(2,3)) # 2x3 가상화면 분할
for(i in 1:3) {
  boxplot(myds[,i], main=colnames(myds)[i])
}
```