

# R프로그래밍및실습 퀴즈\_3 수강생 답안내용

Jinseog Kim  
Dongguk University  
jskim1986@gmail.com

2018-05-29

## Contents

|             |    |
|-------------|----|
| 1 김탄 - 0    | 2  |
| 2 이재용 - 0   | 2  |
| 3 송희연 - 5   | 2  |
| 4 김정렬 - 0   | 3  |
| 5 이장영 - 0   | 4  |
| 6 이승진 - 0   | 4  |
| 7 유태현 - 1   | 4  |
| 8 이현주 - 0   | 5  |
| 9 허선민 - 5   | 6  |
| 10 김범석 - 2  | 6  |
| 11 이향림 - 0  | 6  |
| 12 이진형 - 10 | 7  |
| 13 박하영 - 7  | 7  |
| 14 진민근 - 10 | 9  |
| 15 정의주 - 3  | 10 |
| 16 엄준호 - 0  | 12 |
| 17 김도환 - 0  | 12 |
| 18 최주원 - 0  | 13 |
| 19 오형근 - 5  | 13 |
| 20 유현수 - 0  | 14 |
| 21 신세희 - 0  | 14 |
| 22 정석환 - 2  | 15 |
| 23 추민환 - 10 | 15 |

```
# 읽고자 하는 파일 이름
pop_f <- "건강보험인구현황.csv"
acc_f <- "2015_사망사고.csv"
cars_f <- "자동차등록대수(11년-15년).csv"
pop_2015 <- read.csv(pop_f, fileEncoding = "CP949")
acc_2015 <- read.csv(acc_f, fileEncoding = "CP949")
```

## 1 김탄 - 0

```
# <<<<===== pop_f <- read.csv("건강보험인구현황.csv")
# <<<<===== acc_f <- read.csv("2015_사망사고.csv")
# <<<<===== cars_f <- read.csv("자동차등록대수(11년-15년).csv")
# <<<<===== pop_2015 <- read.csv("건강보험인구현황.csv", fileEncoding = "CP949")
# <<<<===== acc_2015 <- read.csv("2015_사망사고.csv", fileEncoding = "CP949")
# <<<<===== cars_2015 <- read.csv("자동차등록대수(11년-15년).csv", fileEncoding = "CP949")
```

## 2 이재용 - 0

```
# m1 <- merge(pop_2015$ 시도, pop_2015$ 인구수, by.x=" 시도", by.y=" 인구수")
# Error in fix.by(by.x, x) : 'by' must specify a uniquely valid column

# subset(pop_2015, select=c(" 시도", " 인구수"))
#<<<< x <- subset(pop_2015, select=c(" 시도", " 인구수"))
#<<<< y <- merge(x, freq_accidental, by.x=" 인구수", by.y=" 사건수")
# Error in as.data.frame(y) : 객체 'freq_accidental'를 찾을 수 없습니다

#names(y) <- c(" 시도", " 인구수", " 시도2", " 사건수")
#z <- subset(y, select=c( 시도, 인구수, 사건)수)
```

## 3 송희연 - 5

```
# 시도별 인구대비_사건사고 구하기
# 1) 데이터 파일 건강보험인구현황.csv/2015_사망사고.csv/자동차등록대수(11년-15년).csv를 읽어 데이터프레임 pop_2015와 acc_2015을 만든다
pop_2015 <- read.csv(pop_f, fileEncoding = "CP949")
acc_2015 <- read.csv(acc_f, fileEncoding = "CP949")

# 필요없는 코드이며 각 코드에 대한 설명 없음

cars_2015 <- read.csv(cars_f, fileEncoding = "CP949")
cars_2015$구분 <- factor(as.character(cars_2015$구분), levels=c("서울", "부산", "대구", "인천", "광주", "대전", "울산", "세종", "경기", "충청", "전남", "전북", "제주"))
cars_2015 <- subset(cars_2015, select=c("구분", "X2015"))
acc_2015$발생지시도 <- factor(as.character(acc_2015$발생지시도), levels=c("서울", "부산", "대구", "인천", "광주", "대전", "울산", "세종", "경기", "충청", "전남", "전북", "제주"))

# 이 코드를 이해하고 작성하였는지 의문 ???
pop_2015$시도 <- as.character(pop_2015$시도)
pop_2015$시도 <- gsub("특별시|광역시|특별자치시|특별자치도|도", "", pop_2015$시도)
do_idx <- nchar(pop_2015$시도)
```

```
pop_2015$ 시도[do_idx] <- paste0(substring(pop_2015$ 시도[do_idx], 1, 1), substring(pop_2015$ 시도[do_idx], 3, 3))
pop_2015$ 시도 <- factor(pop_2015$ 시도, levels=unique(pop_2015$ 시도))
```

#2) 시도별 총인구수(단위 100만)를 구한다.

```
pop_2015$ 인구수 <- pop_2015$ 인구수/1000000
(pop_sido <- aggregate(인구수~시도, data=pop_2015, sum))
```

#3) 사망사고중사고유형\_대분류가 '차대차'이고 컬럼명이 '발생지시도', '사망자수'만을 추출하여 acc\_2015\_차대차에 저장한다.  
acc\_2015\_차대차 <- subset(acc\_2015, 사고유형\_대분류=='차대차', select=c('발생지시도', '사망자수'))  
head(acc\_2015\_차대차)

#4) acc\_2015\_차대차에서 총 발생건수 및 총 사망자수를 구한다.

```
acc_n <- aggregate(사망자수~발생지시도, data=acc_2015_차대차, length)
acc_sum <- aggregate(사망자수~발생지시도, data=acc_2015_차대차, sum)
acc_sido <- merge(acc_n, acc_sum, by="발생지시도", sort = F)
names(acc_sido)[-1] <- c("사고건수", "사망자수")
acc_sido
```

#5) 시도별 총인구수와 시도별 사망사고 중 차대차로 인한 발생건수 및 사망자수 데이터를 시도별로 병합한다.

```
acc_sido <- merge(pop_sido, acc_sido, by.y="발생지시도", by.x="시도", sort = F)
acc_sido
```

#6) 시도별 총인구수와 시도별 사망사고 중 차대차로 인한 발생건수 및 사망자수 데이터를 시도별로 병합한다.

```
acc_sido <- merge(acc_sido, cars_2015, by.x="시도", by.y="구분", sort = F)
names(acc_sido)[5] <- "자동차수"
acc_sido$자동차수 <- acc_sido$자동차수/1000000
```

#7) 시도별 인구대비\_사고건수를 구한다.

```
acc_sido$인구대비_사고건수 <- acc_sido$사고건수/acc_sido$인구수
barplot(acc_sido$인구대비_사고건수, names.arg = acc_sido$시도, main="시도별 인구대비 사고건수", las=2)
```

## 4 김정렬 - O

```
# , fileEncoding = "CP949") 추가
pop_2015 <- read.csv("건강보험인구현황.csv", fileEncoding = "CP949")
acc_2015 <- read.csv("2015_사망사고.csv", fileEncoding = "CP949")
# 필요없는 코드
cars_2015 <- read.csv("자동차등록대수(11년-15년).csv")
```

# 코드에 대한 설명 없음

```
a <- aggregate(pop_2015$인구수, list(시도 = pop_2015$시도), sum)
names(a)[2] <- c("인구수")
d <- aggregate(acc_2015$발생분, list(시도 = acc_2015$발생지시도), sum)
names(d)[2] <- ("사고수")
```

## 5 이장영 - 0

```
p <- aggregate(인구수~시도, data=pop_2015, FUN=sum)
a <- table(acc_2015$발생지시도)
a_dt <- data.frame(a)
names(a_dt) <- c("시도", "사고수")

#p객체와 a_dt객체를 "시도"별로 merge 시킨다.
merge(a_dt, p, by="시도")
```

## 6 이승진 - 0

```
## , fileEncoding = "CP949")추가
pop_f <- read.csv("건강보험인구현황.csv", fileEncoding = "CP949")
acc_f <- read.csv("2015_사망사고.csv", fileEncoding = "CP949")
cars_f <- read.csv("자동차등록대수(11년-15년).csv", fileEncoding = "CP949")

#getwd()

# 필요없는 코드
#class(acc_f$발생지시도)
freq_mm <- table(acc_f$발생지시도)
#freq_mm
#barplot(freq_mm)

#class(pop_f)

# 에러발생
m1 <- merge(freq_mm, y, by.x = "발생지시도", by.y = "시도")
## Error in as.data.frame(y) : 객체 'y'를 찾을 수 없습니다

# ???
tapply(pop_f$시도,pop_f$인구수,fun=sum)

#발생지시도 사고건수를 인구수 데이터로 나눠서 표작성
```

## 7 유태현 - 1

```
## , fileEncoding = "CP949") 추가
pop_2015 <- read.csv("건강보험인구현황.csv", fileEncoding = "CP949")
acc_2015 <- read.csv("자동차등록대수(11년-15년).csv", fileEncoding = "CP949")
## 필요없는 코드 아래...
cars_2015 <- read.csv("2015_사망사고.csv", fileEncoding = "CP949")
cars_2015$구분 <- factor(as.character(cars_2015$구분),
                        levels=c("서울", "부산", "대구", "인천", "광주", "대전",
                                   "울산", "세종", "경기", "강원", "충북", "충남", "전북", "전남",
                                   "경북", "경남", "제주"))

cars_2015 <- subset(cars_2015, select=c("구분", "X2015"))
```

```

acc_2015$발생지 시도 <- factor(stringr::str_trim(as.character(acc_2015$발생지 시도)),
                             levels=c("서울", "부산", "대구", "인천", "광주", "대전",
                                       "울산", "세종", "경기", "강원", "충북", "충남", "전북", "전남",
                                       "경북", "경남", "제주"))

pop_2015$시도 <- as.character(pop_2015$시도)

pop_2015$시도 <- gsub("특별시|광역시|특별자치시|특별자치도|도", "", pop_2015$시도)

do_idx <- nchar(pop_2015$시도)==3

pop_2015$시도[do_idx] <- paste0(substring(pop_2015$시도[do_idx], 1, 1),
                               substring(pop_2015$시도[do_idx], 3, 3))
pop_2015$시도 <- factor(pop_2015$시도, levels=unique(pop_2015$시도))

head(pop_2015)

pop_2015$시도 <- factor(as.character(pop_2015$시도),
                       levels=c("서울특별시", "부산광역시", "대구광역시", "인천광역시", "광주광역시",
                                "대전광역시", "울산광역시", "세종특별자치시", "경기도", "강원도", "충청북도", "충청남도", "전라북도", "전라남도",
                                "경상북도", "경상남도", "제주특별자치도"))

```

## 8 이현주 - 0

```

# <<===== pop_f <- "http://dataming.dongguk.ac.kr/lectures/2018-1/R/건강보호인구현황.csv"
# <<===== acc_f <- "http://dataming.dongguk.ac.kr/lectures/2018-1/R/2015_사망사고.csv"
# <<===== cars_f <- "http://dataming.dongguk.ac.kr/lectures/2018-1/R/자동차등록대수(11년-15년).csv"
# <<===== pop_2015 <- read.csv(pop_f, fileEncoding = "CP949")
# <<===== acc_2015 <- read.csv(acc_f, fileEncoding = "CP949")
# <<===== cars_2015 <- read.csv(cars_f, fileEncoding = "CP949")
# <<===== read.csv("C:/users/stat/downloads/자동차등록대수(11년-15년).csv", fileEncoding="CP949")

# <<===== pop_f <- "http://dataming.dongguk.ac.kr/lectures/2018-1/R/건강보험인구현황.csv"
# <<===== acc_f <- "http://dataming.dongguk.ac.kr/lectures/2018-1/R/2015_사망사고.csv"
# <<===== cars_f <- "http://dataming.dongguk.ac.kr/lectures/2018-1/R/자동차등록대수(11년-15년).csv"
# <<===== pop_2015 <- read.csv(pop_f, fileEncoding = "CP949")
# <<===== acc_2015 <- read.csv(acc_f, fileEncoding = "CP949")
# <<===== cars_2015 <- read.csv(cars_f, fileEncoding = "CP949")

# <<===== pop_f <- "http://dataming.dongguk.ac.kr/lectures/2018-1/R/건강보험인구현황.csv"
# <<===== acc_f <- "http://dataming.dongguk.ac.kr/lectures/2018-1/R/2015_사망사고.csv"
# <<===== cars_f <- "http://dataming.dongguk.ac.kr/lectures/2018-1/R/자동차등록대수(11년-15년).csv"
# <<===== traffic_acc <- read.csv(pop_f, fileEncoding = "CP949")

# <<<< C:\Users\stat\Documents

# <<===== pop_2015 <- read.csv("http://dataming.dongguk.ac.kr/lectures/2018-1/R/건강보호인구현황.csv",fileEncoding = "CP949")
# <<===== titanic <- read.table("titanic.txt", header=T)
# <<===== head(titanic)

```

## 9 허선민 - 5

```
pop_f<-"C:/Users/stat/Downloads/건강보험인구현황.csv"
acc_f<-"C:/Users/stat/Downloads/2015_사망사고.csv"
cars_f<-"C:/Users/stat/Downloads/자동차등록대수(11년-15년).csv"
pop_2015<-read.csv(pop_f, fileEncoding = "CP949")
acc_2015<-read.csv(acc_f, fileEncoding = "CP949")
cars_2015<-read.csv(cars_f, fileEncoding = "CP949")
pop_2015
acc_2015
cars_2015

#시도별 총 인구수를 구한 것
pop_2015$인구수 <- pop_2015$인구수/1000000
#시도별 인구수의 합을 출력한 것을 pop_sido에 저장함
pop_sido <- aggregate(인구수~시도, data=pop_2015, sum)
#acc_2015자료의 '발생지시도'와 '사망자수'만 뽑아서 자료를 보는 것을 acc_2015에 저장한다.
# ???
acc_2015_차대차 <- subset(acc_2015, 사고유형_대분류=='차대차', select=c('발생지시도', '사망자수'))

head(acc_2015_차대차)
acc_n <- aggregate(사망자수~발생지시도, data=acc_2015_차대차, length) #acc_2015 자료의 발생지시도의 사망자수의 길이를 출력함
acc_sum <- aggregate(사망자수~발생지시도, data=acc_2015_차대차, sum) #acc_2015 자료의 발생지시도의 사망자수의 합을 출력함
acc_sido <- merge(acc_n, acc_sum, by="발생지시도", sort = F) #위의 두개의 자료를 "발생지시도"를 기준으로 병합함
names(acc_sido)[-1] <- c("사고건수", "사망자수") # <<<<<< "변수의 이름을 "사고건수", "사망자수"로 바꾼다

acc_sido
```

## 10 김범석 - 2

```
pop_시도별인구 <-subset(pop_2015, subset= 인구수 >0 , select=c(시도,인구수))
# 시도별 인구를 pop_2015 에서 추출합니다.
acc_사고수 <-subset(acc_2015, subset= 사망자수 >=0 , select=c(사망자수,중상자수,사상자수,경상자수))
# 2015년 사고수= (사망자수, 중상자수, 사상자수, 경상자수) 를 acc_2015에서 추출합니다.

# <<<< 그 이후 사고수 = 사망자수+중상자수+사상자수+경상자수 를 구한 뒤
# <<<< merge 함수와 aggregate 함수 를 통해 pop_시도별인구와 acc_사고수를 합칩니다.
# <<<< merge 함수를 통해 합친 뒤 사고수/인구수를 구하여 변수에 추가합니다.
```

## 11 이향림 - 0

```
# <<===== pop_f <- "C:/Users/stat/Desktop/건강보험인구현황.csv"
# <<===== acc_f <- "C:/Users/stat/Desktop/2015_사망사고.csv"
# <<===== cars_f <- "C:/Users/stat/Desktop/자동차등록대수(11년-15년).csv"
# <<===== pop_2015 <- read.csv(pop_f, fileEncoding = "CP949")
# <<===== acc_2015 <- read.csv(acc_f, fileEncoding = "CP949")
# <<===== cars_2015 <- read.csv(cars_f, fileEncoding = "CP949")
```

## 12 이진형 - 10

```
# <<===== pop_2015<-read.csv("C:/Users/stat/Downloads/건강보험인구현황.csv",fileEncoding="CP949")
# <<===== acc_2015<-read.csv("C:/Users/stat/Downloads/2015_사망사고.csv",fileEncoding="CP949")
# <<===== cars_2015<-read.csv("C:/Users/stat/Downloads/자동차등록대수(11년-15년).csv",fileEncoding="CP949")
pop_2015$ 시도 <- as.character(pop_2015$ 시도)
pop_2015$ 시도 <- gsub("특별시|광역시|특별자치시|특별자치도", "", pop_2015$ 시도)
do_idx <- nchar(pop_2015$ 시도)==3
pop_2015$ 시도[do_idx] <- paste0(substring(pop_2015$ 시도[do_idx], 1, 1),
                                substring(pop_2015$ 시도[do_idx], 3, 3))

pop_2015$ 시도 <- factor(pop_2015$ 시도, levels=unique(pop_2015$ 시도))
pop_2015$ 인구수 <- pop_2015$ 인구수/1000000

(pop_sido <- aggregate(인구수~시도, data=pop_2015, sum))

#100만단위의 인구수를 구하기 위해서 인구수에 100만을 나눈 값을 인구수로 놓고
#시도별로 합치기 위해 aggregate(합치는 자료,데이터, 합치기)를 실행한다.
acc_2015_차대차<-subset(acc_2015, 사고유형_대분류=='차대차',select= c('발생지시도','사망자수'))

#일부만 추출하기 위해 subset(자료, 조건)의 함수를 사용한다. '발생지시도','사망자수'를 추출하기 위해서 select함수를 사용하고
#'발생지시도'와 '사망자수'를 벡터로 표시한다.
acc_n <- aggregate(사망자수~발생지시도, data=acc_2015_차대차, length)
acc_sum <- aggregate(사망자수~발생지시도, data=acc_2015_차대차, sum)
acc_sido <- merge(acc_n, acc_sum, by="발생지시도")
names(acc_sido)[-1] <- c("사고건수", "사망자수")
acc_sido

#시도별 사고건수와 사망자수를 구하기 위해 acc_n과 acc_sum이라는 객체를 만든다
#acc_n은 acc_2015_차대차라는 자료에서 발생지시도에 따른 사망자수의 길이(length)를 표시한다
#acc_sum은 acc_2015_차대차라는 자료에서 발생지시도에 따른 사망자수의 합(sum)를 표시한다
#그리고 시도별로 묶기 위해서 merge(x,y,by="")함수를 사용한다
#이 함수는 발생지시도에 따른 acc_n과 acc_sum을 합치는 것이다.
#이렇게 하면 이름이 발생지시도 사망자수.x 사망자수.y로 나온다
#names()[-1]을 통해서 첫번째 발생지시도를 제외한 두개의 변수 이름을 사고건수와 사망자수로 변환한다
#그리고 acc_sido를 출력한다.

acc_sido<- merge(pop_sido, acc_sido, by.y="발생지시도", by.x="시도")
acc_sido

#acc_sido 라는 객체에 pop_sido와 acc_sido데이터의 발생지시도와 시도가 같은 것끼리 묶는다.
#그리고 acc_sido_를 출력한다
acc_sido$인구대비_사고건수 <- acc_sido$사고건수/acc_sido$인구수

#acc_sido$인구대비_사고건수를 사고건수를 인구수로 나눈 것에 대응시키고
#시도별 인구수, 사고수, 인구대비사고건수만 남기고 다 제거하여 표를 만든다
```

## 13 박하영 - 7

```
# <<===== getwd()
# 데이터 파일을 읽기위해 우선 컴퓨터 저장디렉토리를 찾는다.
```

```

# <===== pop_2015 <- read.csv("건강보험인구현황.csv", fileEncoding="CP949")
# <===== acc_2015 <- read.csv("2015_사망사고.csv", fileEncoding="CP949")
# <===== cars_2015 <- read.csv("자동차등록대수(11년-15년).csv", fileEncoding="CP949")

acc_2015$발생지시도 <- factor(as.character(acc_2015$발생지시도),
                             levels=c("서울", "부산", "대구", "인천", "광주", "대전",
                                       "울산", "세종", "경기", "강원", "충북", "충남", "전북", "전남",
                                       "경북", "경남", "제주"))
# acc_2015 데이터프레임에서 발생지시도를 "서울", "부산", "대구", "인천", "광주", "대전",
#"울산", "세종", "경기", "강원", "충북", "충남", "전북", "전남", "경북", "경남", "제주" 순으로
# 하고 또 해당된 시도에 맞는 데이터들이 오도록 하기 위해 levels를 사용한다.
# 이 모든것을 하기 위해 우선 범주형변수인 acc_2015$발생지시도를 문자형벡터로 전환하는것이 필요하다.
pop_2015$시도 <- as.character(pop_2015$시도)

# pop_2015$시도를 문자형 벡터로 바꾼다.
pop_2015$시도 <- gsub("특별시|광역시|특별자치시|특별자치도|도", "", pop_2015$시도)

# gsub는 전역을 교체하는 것으로 pop_2015$시도에서 뒷자리의 특별시, 광역시, 특별자치시, 특별자치도를 뺀다.
do_idx <- nchar(pop_2015$시도)==3

# nchar(x) 는 문자형 벡터x의 구성요소의 개수를 구하는 것으로 여기선 pop_2015$시도의 개수 즉, 단어의 수가 3개인 것을
# do_idx로 저장한다. 이때 결과는 TRUE 또는 FALSE인논리형으로 나온다.
pop_2015$시도[do_idx] <- paste0(substring(pop_2015$시도[do_idx], 1, 1), substring(pop_2015$시도[do_idx], 3, 3))

# paste0은 각각의 원소를 공백없이 이어주는 함수이고, substring은 문자역의 일부를 추출해 주는 것이다.
# 즉, pop_2015$시도[do_idx]의 첫번째 단어와, pop_2015$시도[do_idx]의 3번째 단어를 공백없이 이어준다.
pop_2015$시도 <- factor(pop_2015$시도, levels=unique(pop_2015$시도))

# 마지막으로 pop_2015$시도 를 중복없이 한번씩만 나타내며 범주형으로 만들어 준다.
pop_2015$인구수 <- pop_2015$인구수/1000000 #인구수를 100만 단위로 한다.
pop_시도 <- aggregate(pop_2015$인구수, list(pop_2015$시도), sum)

# 시도를 기준으로 pop_2015의 인구수의 합을 pop_시도 에 저장한다.
names(pop_시도)[1:2] <- c("시도", "인구수")

# 만들어진 pop_시도의 첫번째와 두번째의 변수명을 각각 시도, 인구수로 한다.
acc_2015_사고수 <- aggregate(acc_2015$사망자수, list(acc_2015$발생지시도), length)

#발생지시도를 기준으로 데이터프레임 acc_2015의 사망자수의 길이(개수)를 통해 사고수를 구한다.
#이를 acc_2015_사고수로 저장한다.

names(acc_2015_사고수)[1:2] <- c("시도", "사고건수")

# 만들어진 acc_2015_사고수의 첫번째와 두번째의 변수명을 각각 시도, 사고건수로 한다.
acc_chart <- merge(pop_시도, acc_2015_사고수, by="시도")

# 위에서 만든 pop_시도와 acc_2015_사고수를 병합한다. 이때 두 데이터프레임의 공통열인 시도를 기준으로 병합한다.
acc_chart$인구대비사고건수 <- acc_chart$사고건수/acc_chart$인구수

# 시도별 인구대비 사고건수는 시도별 사고건수/시도별 인구수로 구한다.
# 따라서 pop_시도와 acc_2015_사고수를 병합한 데이터프레임인 acc_chart를 이용한다.

#####>>>????????????????????

```



```
acc_chart <- merge(acc_chart, acc_chart$인구대비사고건수, by="시도")
```

```
# 마지막으로 pop_시도와 acc_2015 사고수를 병합한 데이터프레임인 acc_chart에  
# 시도별인구대비사고건수를 병합한 것을 다시 acc_chart로 만든다.
```

## 14 진민곤 - 10

```
setwd("C:/Users/user/Desktop/R_Programing")  
pop_2015 <- read.csv("건강보험인구현황.csv", fileEncoding = "CP949")  
acc_2015 <- read.csv("2015_사망사고.csv", fileEncoding = "CP949")  
cars_2015 <- read.csv("자동차등록대수(11년-15년).csv", fileEncoding = "CP949")  
head(pop_2015)  
head(acc_2015)  
head(cars_2015)  
  
cars_2015$구분 <- factor(as.character(cars_2015$구분),  
                        levels=c("서울", "부산", "대구", "인천", "광주", "대전",  
                                "울산", "세종", "경기", "강원", "충북", "충남", "전북", "전남",  
                                "경북", "경남", "제주"))  
  
#cars_2015$구분에("서울", "부산", "대구", "인천", "광주", "대전", "울산", "세종", "경기", "강원",  
#"충북", "충남", "전북", "전남", "경북", "경남", "제주")값의 범주형 변수를 저장  
  
cars_2015 <- subset(cars_2015, select=c("구분", "x2015"))  
  
#car_2015에 cars_2015에서 x2015를 만족하는 값을 저장  
# <<<< install.packages("stringr")  
  
acc_2015$발생지시도 <- factor(stringr::str_trim(as.character(acc_2015$발생지시도)),  
                              levels=c("서울", "부산", "대구", "인천", "광주", "대전",  
                                        "울산", "세종", "경기", "강원", "충북", "충남", "전북", "전남",  
                                        "경북", "경남", "제주"))  
  
#위의 범주형 변수를 문자벡터로 변환하여 저장  
pop_2015$시도 <- as.character(pop_2015$시도)  
  
#pop_2015$시도를 문자벡터로 변환하여 저장  
  
pop_2015$시도 <- gsub("특별시|광역시|특별자치시|특별자치도|도", "", pop_2015$시도)  
  
#pop_2015에 특별시|광역시|특별자치시|특별자치도|도 의 명칭을 ""로 변환(특별시|광역시|특별자치시|특별자치도|도 를 없애줌)  
do_idx <- nchar(pop_2015$시도)==3  
  
#pop_2015$시도에 3을 갖는 것들을 do_idx에 저장  
pop_2015$시도[do_idx] <- paste0(substring(pop_2015$시도[do_idx], 1, 1), substring(pop_2015$시도[do_idx], 3, 3))  
  
#substring(pop_2015$시도[do_idx], 1, 1을 연결, pop_2015  
  
pop_2015$시도 <- factor(pop_2015$시도, levels=unique(pop_2015$시도))  
  
(pop_sido <- aggregate(인구수~시도, data=pop_2015, sum))
```

```

pop_2015$인구수 <- pop_2015$인구수/1000000

#pop_2015$인구수에 pop_2015$인구수*1000000을나눈 값을 저장
(pop_sido <- aggregate(인구수~시도, data=pop_2015, sum))

#데이터를 요약해서(인구수~시도를, data=pop_2015변수로하여, 합하라) pop_sido에 저장
acc_2015_차대차 <- subset(acc_2015, 사고유형_대분류=='차대차', select=c('발생지시도', '사망자수'))

#acc_2015_차대차에 acc_2015_차대차에서 사고유형_대분류=='차대차'를 만족하는 발생지시도와 사망자수만 저장하
acc_n <- aggregate(사망자수~발생지시도, data=acc_2015_차대차, length)

#데이터를 요약해서(사망자수~발생지시도를, data=acc_2015_차대차변수로하여, 데이터길이가같게하라) acc_n에 저장
acc_sum <- aggregate(사망자수~발생지시도, data=acc_2015_차대차, sum)

#데이터를 요약해서(사망자수~발생지시도를, data=acc_2015_차대차변수로하여, 합하라) acc_sum에 저장
acc_sido <- merge(acc_n, acc_sum, by="발생지시도", sort = F)

#acc_n, acc_sum에서 발생지시도를 갖는 값들을 병합하여 acc_sido에 저장
names(acc_sido)[-1] <- c("사고건수", "사망자수")

#acc_sido의[-1]번째 변수명을 사고건수 사망자수로 저장
acc_sido <- merge(pop_sido, acc_sido, by.y="발생지시도", by.x="시도", sort = F)

#acc_sido에 합하여 저장하라 pop_sido, acc_sido의 행으로 발생지시도 와 열로 시도의 값을 함께 같은 값들을 병합하라
acc_sido <- merge(acc_sido, cars_2015, by.x="시도", by.y="구분", sort = F)

#acc_sido에 합하여 저장하라 acc_sido, cars_2015의 시도 와 구分的 값을 함께 같은 값을 병합하라
names(acc_sido)[5] <- "자동차수"

#acc_sido의 5번째 object에 자동차수로 이름을 부여하라
acc_sido$자동차수 <- acc_sido$자동차수/1000000

#acc_sido$자동차수에acc_sido$자동차수나누기1000000한 값을 저장하라
acc_sido$인구대비_사고건수 <- acc_sido$사고건수/acc_sido$인구수

#acc_sido$인구대비_사고건수에 acc_sido$사고건수 나누기 acc_sido$인구수를 저장하라
barplot(acc_sido$인구대비_사고건수, names.arg = acc_sido$시도, main="시도별 인구대비 사고건수", las=2)

#막대그래프 acc_sido의 인구대비_사고건수를 막대그래프로 나타내어라

```

## 15 정의주 - 3

```

# 틀 - 시도별 인구대비_사고건수를 사고건수/시도별인구 로 나오게 한다.
# 인구대비_사망자수를 사망자수/인구 로 나오게 한다.
# 사고건수대비_사망자수를 사망자수/사고건수로 나오게 한다.

pop_f <- read.csv("건강보험인구현황.csv", fileEncoding = "CP949")
acc_f <- read.csv("2015_사망사고.csv", fileEncoding = "CP949")
cars_f <- read.csv("자동차등록대수(11년-15년).csv", fileEncoding = "CP949")

# 건강보험인구현황을 pop_f에 저장. 2015_사망사고를 acc_f에 저장. 자동차등록대수를 cars_f 에 저장.

```

```

acc_n <- aggregate(사망자수~발생지시도, data=acc_f_차대차, length)

#사망자수 부터 발생지시도에 대한 acc_f_차대차 값(갯수)을 acc_n에 넣는다.
#length는 갯수 (n)을 의미한다.
acc_sum <- aggregate(사망자수~발생지시도, data=acc_f_차대차, sum)

#사망자수 부터 발생지시도에 대한 acc_f_차대차 값(총합)을 acc_sum에 넣는다.
#sum은 총 합을 의미한다.
acc_sido <- merge(acc_n, acc_sum, by="발생지시도", sort = F)

#acc_n과 acc_sum을 결합(merge)한다. 배열(sort)은 하지 않음.
subset(acc_sido, select = c("사고건수", "사망자수"))

#acc_sido의 이름을 사고건수와 사망자수만 나오게 한다.
#차대차에서 총 발생건수 및 총 사망자수를 나타냄.
acc_sido

pop_f$인구수 <- pop_f$인구수/1000000
(pop_sido <- aggregate(인구수~시도, data=pop_f, sum))

#시도별 인구수를 단위를 100만으로 지정해서 구한다.
acc_sido <- merge(pop_sido, acc_sido, by.x="시도", by.y="발생지시도", sort = F)
acc_sido

#시도별 총인구수와 시도별 사망사고에 대한 차대차로 인한 발생건수 및 사망자수를
#데이터 시도별로 병합한 것.
#pop_sido와 acc_sido를 병합.

acc_sido <- merge(acc_sido, cars_f, by.x="시도", by.y="구분", sort = F)
names(acc_sido)[5] <- "자동차수"
acc_sido$자동차수 <- acc_sido$자동차수/1000000

#acc_sido 에 acc_sido와 cars_f를 병합하여 저장.
#(acc_sido의)이름이 5번째만 나오게 하고 이 이 5번째에 자동차수 라고 저장.
#시도별 총인구수와 시도별 사망사고를 차대차로 인한 발생건수, 사망자수 데이터를 병합.

acc_sido$인구대비_사고건수 <- acc_sido$사고건수/acc_sido$인구수
acc_sido$인구대비_사망자수 <- acc_sido$사망자수/acc_sido$인구수
acc_sido$사고건수대비_사망자수 <- acc_sido$사망자수/acc_sido$사고건수
acc_sido$자동차수대비_사망자수 <- acc_sido$사망자수/acc_sido$자동차수

#시도별 인구대비 사고건수와 인구대비 사망자수 사고건수대비 사망자수를
#acc_sido$인구대비_사고건수
#acc_sido$인구대비_사망자수
#acc_sido$사고건수대비_사망자수
#acc_sido$자동차수대비_사망자수 에 저장한다.

acc_sido$인구대비_사고건수
acc_sido$인구대비_사망자수
acc_sido$사고건수대비_사망자수
acc_sido$자동차수대비_사망자수

```

## 16 엄준호 - 0

```
# ????  
pop_f <- read.csv("건강보험인구현황.csv", fileEncoding = "CP949")  
acc_f <- read.csv("2015_사망사고.csv", fileEncoding = "CP949")  
cars_f <- read.csv("자동차등록대수(11년-15년).csv", fileEncoding = "CP949")  
# ????  
pop_2015 <- read.csv("C:/Users/stat/Documents/건강보험인구현황.csv", fileEncoding = "CP949")  
acc_2015 <- read.csv("C:/Users/stat/Documents/2015_사망사고.csv", fileEncoding = "CP949")  
cars_2015 <- read.csv("C:/Users/stat/Documents/자동차등록대수(11년-15년).csv", fileEncoding = "CP949")  
  
# <<<< aggregate를 이용하여 pop_2015에서 지역별 인구를 추출하고  
# <<<< aggregate를 이용하여 acc_2015에서 지역별 사고수를 추출하여  
# <<<< merge를 통하여 추출한 인구와 사고수를 병합하고  
  
x <- merge(pop_2015, acc_2015, by.x="시도", by.y='발생지시도')  
  
# <<<< 사고수/인구  
# <<<< 막대그래프를 이용하여 그래프를 만든다.
```

## 17 김도환 - 0

```
pop_f <- "C:/Users/stat/Documents/kdh/건강보험인구현황.csv"  
acc_f <- "C:/Users/stat/Documents/kdh/2015_사망사고.csv"  
cars_f <- "C:/Users/stat/Documents/kdh/자동차등록대수(11년-15년).csv"  
pop_2015 <- read.csv(pop_f, fileEncoding = "CP949")  
acc_2015 <- read.csv(acc_f, fileEncoding = "CP949")  
cars_2015 <- read.csv(cars_f, fileEncoding = "CP949")  
  
cars_2015$구분 <- factor(as.character(cars_2015$구분),  
                        levels=c("서울", "부산", "대구", "인천", "광주", "대전",  
                                "울산", "세종", "경기", "강원", "충북", "충남", "전북", "전남",  
                                "경북", "경남", "제주"))  
cars_2015 <- subset(cars_2015, select=c("구분", "X2015"))  
acc_2015$발생지시도 <- factor(stringr::str_trim(as.character(acc_2015$발생지시도)),  
                              levels=c("서울", "부산", "대구", "인천", "광주", "대전",  
                                        "울산", "세종", "경기", "강원", "충북", "충남", "전북", "전남",  
                                        "경북", "경남", "제주"))  
pop_2015$시도 <- as.character(pop_2015$시도)  
pop_2015$시도 <- gsub("특별시|광역시|특별자치시|특별자치도", "", pop_2015$시도)  
do_idx <- nchar(pop_2015$시도)==3  
pop_2015$시도[do_idx] <- paste0(substring(pop_2015$시도[do_idx], 1, 1),  
                                substring(pop_2015$시도[do_idx], 3, 3))  
pop_2015$시도 <- factor(pop_2015$시도, levels=unique(pop_2015$시도))
```

## 18 최주원 - 0

```
pop_f <- "C:/Users/stat/Desktop/r프로그램/건강보험인구현황.csv"
acc_f <- "C:/Users/stat/Desktop/r프로그램/2015_사망사고.csv"
cars_f <- "C:/Users/stat/Desktop/r프로그램/자동차등록대수(11년-15년).csv"
pop_2015 <- read.csv(pop_f, fileEncoding = "CP949")
acc_2015 <- read.csv(acc_f, fileEncoding = "CP949")
cars_2015 <- read.csv(cars_f, fileEncoding = "CP949")

pop_2015$인구수 <- pop_2015$인구수/1000000
(pop_sido <- aggregate(인구수~시도, data=pop_2015, sum))

acc_2015_차대차 <- subset(acc_2015, 사고유형_대분류=='차대차', select=c('발생지시도', '사망자수'))
head(acc_2015_차대차)

acc_n <- aggregate(사망자수~발생지시도, data=acc_2015_차대차, length)
acc_sum <- aggregate(사망자수~발생지시도, data=acc_2015_차대차, sum)
acc_sido <- merge(acc_n, acc_sum, by="발생지시도", sort = F)
names(acc_sido)[-1] <- c("사고건수", "사망자수")
acc_sido
```

## 19 오형근 - 5

```
# <<<< 파일열기 객체설정
pop_f<-"건강보험인구현황.csv"
acc_f<-"2015_사망사고.csv"
cars_f<-"자동차등록대수(11년-15년).csv"
pop_2015 <- read.csv(pop_f, fileEncoding = "CP949")
acc_2015<read.csv(acc_f, fileEncoding="CP949")
cars_2015<read.csv(cars_f, fileEncoding="CP949")

# <<<< 인구대비_사고건수구하기
#####
acc_2015_cbc<-subset(acc_2015, 사고유형_대분류=='차대차', select=c('발생지시도', '사망자수'))
acc_n<-aggregate(사망자수~발생지시도, data=acc_2015_cbc, length)
acc_sum<-aggregate(사망자수~발생지시도, data=acc_2015_cbc, sum)
acc_sido<-merge(acc_n, acc_sum, by="발생지시도", sort = F)
names(acc_sido)[-1]<-c("사고건수", "사망자수")
(pop_sido<-aggregate(인구수~시도, data=pop_2015, sum))
##?????

acc_sido <- merge(pop_sido, acc_sido, by.y="발생지시도", by.x="시도", sort = F)
acc_sido <- merge(acc_sido, cars_2015, by.x="시도", by.y="구분", sort = F)
names(acc_sido)[5] <- "자동차수"
acc_sido$자동차수 <- acc_sido$자동차수/1000000
acc_sido$인구대비_사고건수 <- acc_sido$사고건수/acc_sido$인구수
```

## 20 유현수 - 0

```
pop_f <- "http://datamining.doggu.ac.kr/lectures/2018-1/R/건강보험인구현황.csv"
acc_f <- "http://datamining.doggu.ac.kr/lectures/2018-1/R/2015 사망사고.csv"
pop_2015<-read.csv("C:/Users/stat/Desktop/건강보험인구현황.csv",fileEncoding="CP949")
acc_2015<-read.csv("C:/Users/stat/Desktop/자동차등록대수(11년-15년).csv"fileEncoding="cp949")
cars_2015<-read.csv(C:/Users/stat/Desktop/2015_사망사고.csv"fileEncoding="cp949")

# <<<< 시도별 인구대비 사고건수, 인구대비 사망자수, 사고건수대비, 사망자수를 구하라.
# <<<< 서울이 가장 낮은 인구대비 사망자수를 갖고 충남이 가장 높은 인구대비 사망자수를 갖는다.
```

## 21 신세희 - 0

```
pop_f <- ("http://datamining.donggu.ac.kr/lectures/2018-1/R/%EA%B1%B4%EA%B0%95%EB%B3%B4%ED%97%98%EC%9D%B8%EA%B5%AC%ED%98%84%ED%99%A
pop_2015 <- read.csv(pop_f, fileEncoding = "CP949")
acc_2015 <- read.csv("C:/Users/stat/Desktop/2015_사망사고.CSV", fileEncoding = "CP949")
cars_2015 <- read.csv("C:/Users/stat/Desktop/자동차등록대수(11년-15년).CSV")

cars_2015$구분 <- factor(as.character(cars_2015$구분),
                        levels = c("서울", "부산", "대구", "인천", "광주", "대전",
                                   "울산", "세종", "경기", "강원", "충북", "충남", "전북", "전남",
                                   "경북", "경남", "제주"))

cars_2015 <- subset(cars_2015, select=c("구분", "X2015"))

acc_2015$발생지시도 <- factor(stringr::str_trim(as.character(acc_2015$발생지시도)),
                              levels=c("서울", "부산", "대구", "인천", "광주", "대전",
                                       "울산", "세종", "경기", "강원", "충북", "충남", "전북", "전남",
                                       "경북", "경남", "제주"))

pop_2015$시도 <- as.character(pop_2015$시도)
pop_2015$시도 <- gsub("특별시|광역시|특별자치시|특별자치도|도", "", pop_2015$시도)
do_idx <- nchar(pop_2015$시도)==3
pop_2015$시도[do_idx] <- paste0(substring(pop_2015$시도[do_idx], 1, 1), substring(pop_2015$시도[do_idx], 3, 3))
pop_2015$시도 <- factor(pop_2015$시도, levels=unique(pop_2015$시도))

pop_2015$인구수 <- pop_2015$인구수/1000000
(pop_sido <- aggregate(인구수~시도, data=pop_2015, sum))

head(acc_2015)
acc_2015$사고건수 <- acc_2015$사고건수/1000000
(acc_sido <- aggregate(사고건수~시도, data=acc_2015, sum))

class(acc_2015$발생년월일시)
acc_2015$발생월 <- (acc_2015$발생년월일시/%10000)%%100
acc_2015<- table(acc_2015$발생월)
acc_2015

barplot(acc_sido$인구수대비_사고건수, names.arg = acc_sido$시도, main="시도별 인구대비 사건사고", las=2)
```

## 22 정석환 - 2

```
pop_총인구수 <-aggregate(인구수~시도,data=pop_2015,FUN=sum)
# <<<< 시도별 인구수 총합을 요약해서 pop_총인구수에 저장

pop_사고수<-aggregate(사망자수~발생지시도,data=acc_2015,FUN=length)
# <<<< 발생지시도별 사망자수를 길이로 요약해서 pop_사고수에 저장

pop_사고건수 <-merge(pop_총인구수,pop_사고수,by.x="시도",by.y="발생지시도")
# <<<< pop_총인구수 와 pop_사고수의 "시도" 변수와 "발생지시도" 변수를 병합하여서 pop_사고건수에 저장

pop_인구대비_사고건수 <-(pop_사고건수$사망자수/pop_사고건수$인구수)
# <<<< 인구대비 인구수는 인구수/사망자수이다 그값을 pop_인구대비_사고건수에 저장하였다.

table(pop_인구대비_사고건수)
# <<<< 인구대비 사고건수에 대한 표를 만든다.
```

## 23 추민환 - 10

```
pop_2015 <- read.csv("http://datamining.dongguk.ac.kr/lectures/2018-1/R/%EA%B1%B4%EA%B0%95%EB%B3%B4%ED%97%98%EC%9D%B8%EA%B5%AC%ED%98%
acc_2015 <- read.csv("http://datamining.dongguk.ac.kr/lectures/2018-1/R/2015_%EC%82%AC%EB%A7%9D%EC%82%AC%EA%B3%A0.csv")
car_2015 <- read.csv("http://datamining.dongguk.ac.kr/lectures/2018-1/R/%EC%9E%90%EB%8F%99%EC%B0%A8%EB%93%B1%EB%A1%9D%EB%8C%80%EC%88%
head(pop_2015)
head(acc_2015)
head(car_2015)
rm(list=ls())

car_2015$구분 <- factor(as.character(car_2015$구분),
                       levels= c("서울", "부산", "대구", "인천",
                                   "광주", "대전", "울산", "세종",
                                   "경기", "강원", "충북", "충남", "전북",
                                   "전남", "경북", "경남", "제주"))

car_2015 <- subset(car_2015,select = c("구분", "X2015"))

acc_2015$발생지시도 <- factor(stringr::str_trim(as.character(acc_2015$발생지시도)),
                              levels= c("서울", "부산", "대구", "인천",
                                          "광주", "대전", "울산", "세종",
                                          "경기", "강원", "충북", "충남", "전북",
                                          "전남", "경북", "경남", "제주"))

pop_2015$시도 <- as.character(pop_2015$시도)
pop_2015$시도 <- gsub("특별시|광역시|특별자치시|특별자치도", "", pop_2015$시도)
do_idx <- nchar(pop_2015$시도)== 3
pop_2015$시도[do_idx] <- paste0(substring(pop_2015$시도[do_idx], 1, 1),
                               substring(pop_2015$시도[do_idx], 3, 3))
pop_2015$시도 <- factor(pop_2015$시도, levels=unique(pop_2015$시도))

head(pop_2015)
head(acc_2015)
head(car_2015)
```

```

#####

#pop : 인구 , acc : 사고건수 ;;인구대비_사고건수
pop_2015$인구수 <- pop_2015$인구수/1000000
# 데이터프레임 pop_2015 의 변수인 인구수를 1000000으로 나눈다.
pop_sido <- aggregate(pop_2015$인구수, list("시도"=pop_2015$시도), sum)
# 데이터프레임 pop_2015의 인구수를 데이터프레임 pop_2015의 변수인 시도를 기준으로 합한 값으로 pop_sido란 데이터프레임을 만든다.
pop_sido

names(pop_sido)[2] <- "인구수"
# 데이터프레임 pop_sido 의 두번째 열의 변수이름을 인구수로 한다.
pop_sido$인구수 <- sort(pop_sido$인구수, decreasing=T)
# 데이터프레임 pop_sido 의 인구수를 크기순으로 정렬한다.????????????????

head(pop_sido)
#####
acc_2015_n <- aggregate(acc_2015$발생지시도, list(acc_2015$발생지시도), length )
# 데이터프레임 acc_2015의 변수인 발생지시도의 변수값들의 값들의 개수로 acc_2015_n이란 데이터프레임을 만든다.
acc_2015_n

names(acc_2015_n) <- c("시도", "발생건수")
# 데이터프레임 acc_2015_n 의 변수명을 "시도", "발생건수"로 만든다.
head(acc_2015_n)
#####
sol <- acc_2015_n$발생건수/pop_sido$인구수
#데이터프레임 acc_2015의 변수 발생건수를 데이터프레임 pop_sido의 변수 인구수로 나눈 sol이란 벡터값을 만든다.
sol

#####
s <- merge(acc_2015_n, pop_sido, by="시도")
# 데이터프레임 acc_2015_n 과 데이터프레임 pop_sido 를 변수"시도" 를 기준으로 병합한다.
head(s)

#####
s[4] <- sol
#데이터프레임 s의 4번째 열에 벡터값 sol을 합한다.
s

names(s)[4] <- "사고수/인구"
#데이터프레임 s의 4번째 열의 이름을 "사고수/인구"로 한다.
s

```