

제3장 연습문제

문제 1. url이 <https://vincentarelbundock.github.io/Rdatasets/datasets.html> 인 사이트에서 AirPassengers 데이터를 변수 x로 다운받아라. [힌트 : read.csv() 함수 사용]

```
url <- "https://vincentarelbundock.github.io/Rdatasets/csv/datasets/AirPassengers.csv"
x <- read.csv(url)
head(x)
```

결과 :

```
##   X      time value
## 1 1 1949.000    112
## 2 2 1949.083    118
## 3 3 1949.167    132
## 4 4 1949.250    129
## 5 5 1949.333    121
## 6 6 1949.417    135
```

변수 x의 데이터 구조 확인하기.

```
str(x)
```

결과 :

```
## 'data.frame':   144 obs. of  3 variables:
## $ X      : int  1 2 3 4 5 6 7 8 9 10 ...
## $ time : num  1949 1949 1949 1949 1949 ...
## $ value: int  112 118 132 129 121 135 148 148 136 119 ...
```

문제 2. 다운받은 데이터를 /temp 디렉토리에 "AirPassengers.txt" 로 저장하라.

[힌트 : write.table() 함수 사용]

```
setwd("C:/Temp")
write.table(x, "AirPassengers.txt", sep=",")

# 파일이 저장되었는지 확인
list.files(pattern="AirPassengers.txt")
dir(pattern="AirPassengers.txt")
```

결과 :

```
## [1] "AirPassengers.txt"
```

문제 3.

3-1) 다운받은 데이터 세트의 time 열 값을 x1 벡터에 대입하라.

```
x1 <- x$time
head(x1)
```

결과 :

```
## [1] 1949.000 1949.083 1949.167 1949.250 1949.333 1949.417
```

3-2) 다운받은 데이터 세트의 value를 x2 벡터에 대입하라.

```
x2 <- x$value
head(x2)
```

결과 :

```
## [1] 112 118 132 129 121 135
```

3-3) x1 벡터와 x2 벡터를 열 결합하여 y 배열을 생성하라. [cbind() 함수 사용]

```
y <- cbind(x1, x2)
head(y)
```

결과 :

```
##           x1  x2
## [1,] 1949.000 112
## [2,] 1949.083 118
## [3,] 1949.167 132
## [4,] 1949.250 129
## [5,] 1949.333 121
## [6,] 1949.417 135
```

3-4) y 배열의 컬럼 이름을 각각 time과 value로 지정하고, y 배열의 값들을 확인하라. [colnames() 함수 사용]

```
names <- c("time", "value")
colnames(y) <- names
head(y)
```

결과 :

```
##           time value
## [1,] 1949.000    112
## [2,] 1949.083    118
## [3,] 1949.167    132
## [4,] 1949.250    129
## [5,] 1949.333    121
## [6,] 1949.417    135
```

문제 4. [힌트 : 교재 72쪽 참조]

4-1) y 배열에 있는 value 의 총합을 구하라.

```
sum(y[, "value"])
```

결과 :

```
## [1] 40363
```

4-2) y 배열에 있는 value 의 데이터 개수를 구하라.

```
length(y[, "value"])
```

결과 :

```
## [1] 144
```

4-3) y 배열에 있는 value 의 평균을 구하라.

```
mean(y[, "value"])
```

결과 :

```
## [1] 280.2986
```

4-4) y 배열에 있는 value 의 표준편차를 구하라.

```
sd(y[, "value"])
```

결과 :

```
## [1] 119.9663
```

4-5) y 배열의 데이터 세트 요약 정보를 출력하라.

```
summary(y[, "value"])
```

결과 :

##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	104.0	180.0	265.5	280.3	360.5	622.0

