

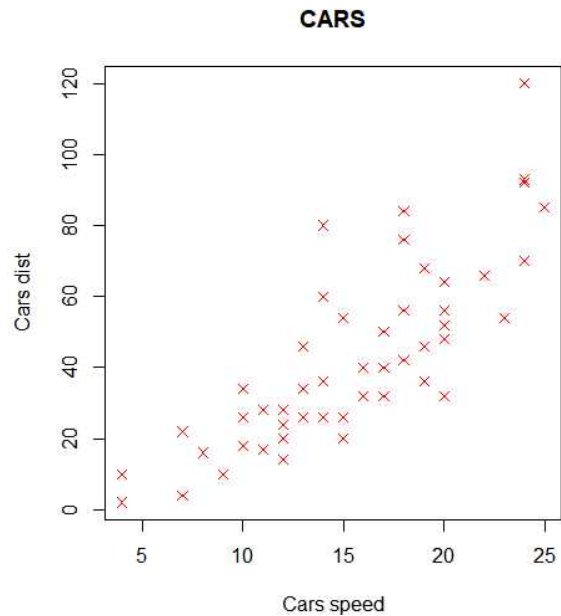
Chapter 16. 연습문제 #1

#1-1

1. R에서 제공하는 cars 데이터셋을 이용해서 speed 와 dist 에 대한 산점도를 그리시오.
(x축이 speed). speed 와 dist (제동거리)에 대한 상관관계를 설명해 보시오

```
sp <- cars$speed
di <- cars$dist
plot(sp,di,
      main = "CARS",
      xlab = "Cars speed",
      ylab = "Cars dist",
      col = "red",
      pch = 4)
```

- 차의 속도에 따라 차의 제동거리가 길어지는 것을 볼 수 있다.

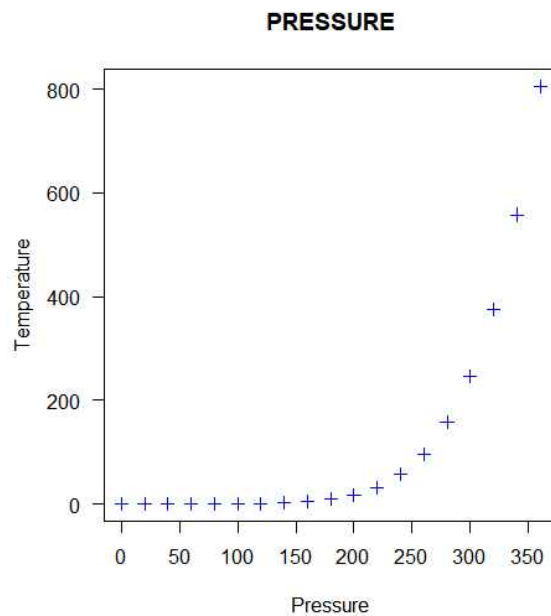


#1-2

2. R에서 제공하는 pressure 데이터셋을 이용해서 temperature 와 pressure에 대한 산점도를 그리시오.
(x축이 temperature). 두 변수간 상관관계를 설명해 보시오

```
pr <- pressure$pressure
te <- pressure$temperature
plot(te,pr,
      main = "PRESSURE",
      xlab = "Pressure",
      ylab = "Temperature",
      col = "blue",
      pch = 3,
      las = 1)
```

- 일정 압력에 도달하면서부터 온도가 높아지는 것을 볼 수 있다.



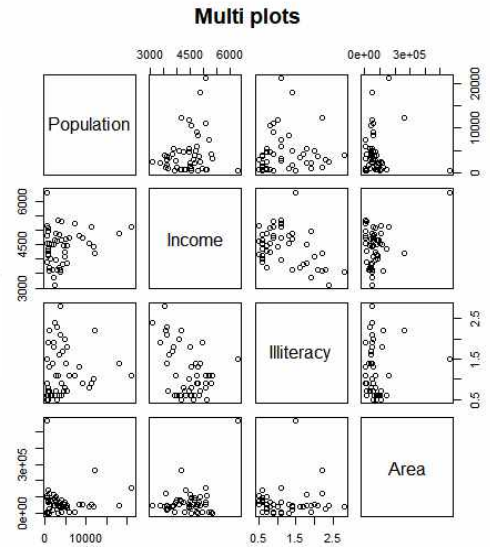
#1-3

3. R에서 제공하는 state.x77 데이터셋에서 Population, Income, Illiteracy, Area 변수간 산점도를 그려 상관관계를 관찰하시오 (pairs() 함수 이용)

```
vars <- c("Population", "Income",
          "Illiteracy", "Area")
target <- state.x77[,vars]
pairs(target,
      main="Multi plots")
```

```
> cor(st$Population, st[c(2,3,8)])
      Income Illiteracy      Area
[1,] 0.2082276 0.1076224 0.02254384
> cor(st$Income, st[c(1,3,8)])
      Population Illiteracy      Area
[1,] 0.2082276 -0.4370752 0.3633154
> cor(st$Illiteracy, st[c(1,2,8)])
      Population      Income      Area
[1,] 0.1076224 -0.4370752 0.07726113
> cor(st$Area, st[c(1,2,3)])
      Population      Income      Illiteracy
[1,] 0.02254384 0.3633154 0.07726113
```

```
st <- data.frame(state.x77)
cor(st$Population, st[c(2,3,8)])
cor(st$Income, st[c(1,3,8)])
cor(st$Illiteracy, st[c(1,2,8)])
cor(st$Area, st[c(1,2,3)])
```

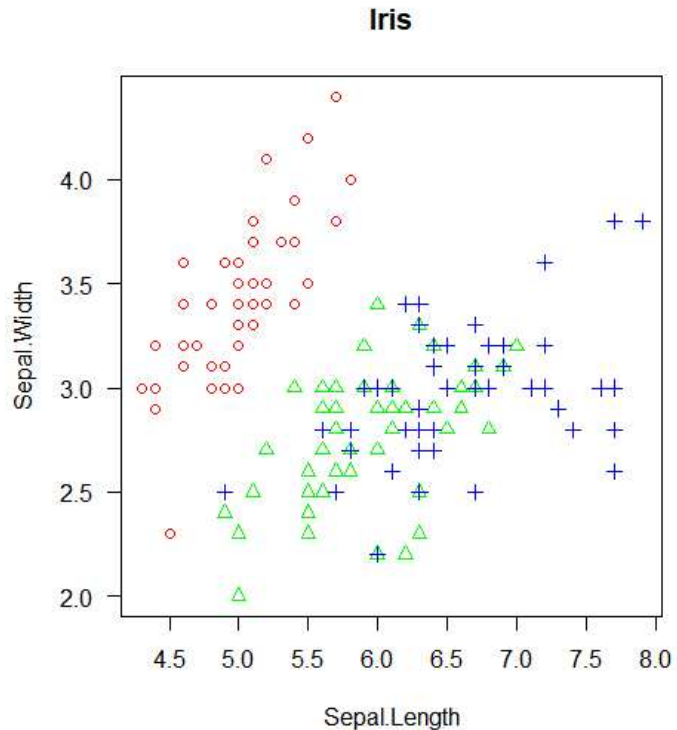


- 상관계수가 보여 주듯이 변수들 간의 관련성이 작기 때문에 뚜렷한 관계를 설명할 수 없다.

#1-4

4. iris 데이터셋에서 Species 정보에 따른 Sepal.Length, Sepal.Width (꽃받침의 길이, 폭)의 분포를 알아 보시오

```
iris1 <- iris[,1:2]
point <- as.numeric(iris$Species)
color <- c("red", "green", "blue")
plot(iris1,
     main="Iris",
     pch=c(point),
     col=color[point],
     las = 1)
```



Chapter 16. 연습문제 #2

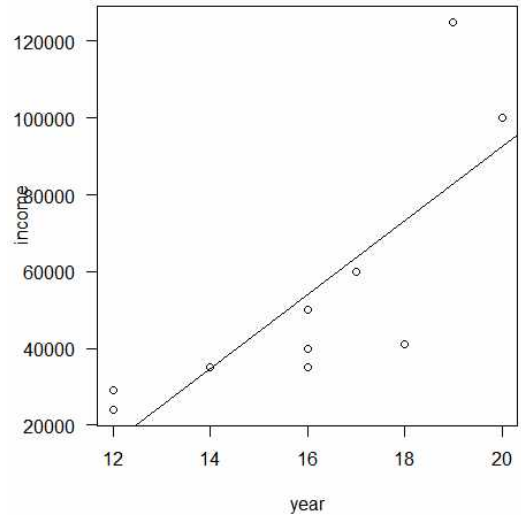
#2-1

1. 다음은 10명의 수입과 교육받은 기간을 조사한 표이다.

수입과 교육기간 사이에 어느 정도 상관관계가 있는지 조사하시오 (산점도, 상관계수 구하기)

```
income=c(125000,100000,40000,35000,41000,
          29000,35000,24000,50000,60000)
year <- c(19,20,16,16,18,12,14,12,16,17)
tbl = data.frame(cbind(income,year))
tbl; class(tbl)
plot(income~year,data=tbl)
res=lm(income~year,data=tbl)
abline(res)
cor(income,year)
```

```
> cor(income,year)
[1] 0.7887259
```



- 대체로 교육받은 기간이 길수록 수입이 크게 나타난다.

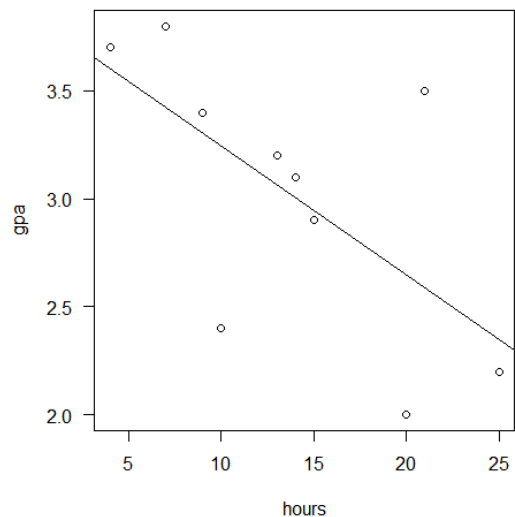
#2-2

2. 다음은 학생 10명의 성적과 TV 시청시간을 조사한 표이다.

성적과 TV시청시간 사이의 상관관계를 조사하시오. (산점도,상관계수 구하기)

```
gpa <- c(3.1,2.4,2.0,3.8,2.2,3.4,2.9,3.2,3.7,3.5)
hours <- c(14,10,20,7,25,9,15,13,4,21)
tbl = data.frame(cbind(gpa,hours))
tbl; class(tbl)
plot(gpa~hours,data=tbl)
res=lm(gpa~hours,data=tbl)
abline(res)
cor(gpa,hours)
```

```
> cor(gpa,hours)
[1] -0.6283671
```



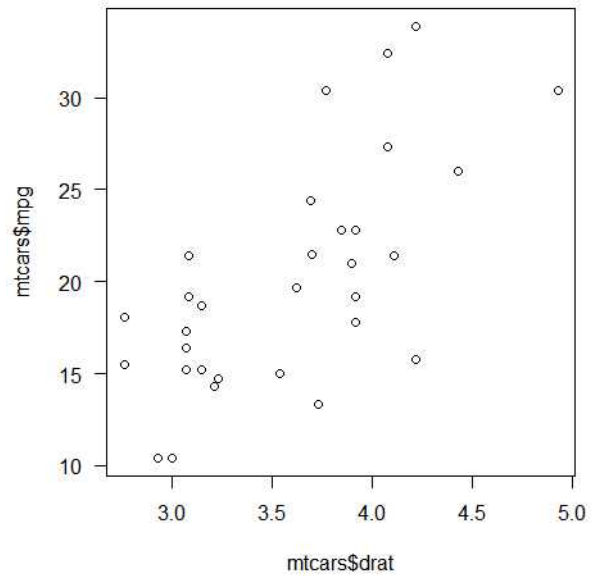
- 대체로 TV시청시간이 길수록 성적이 낮게 나타난다.

#2-3

3. R에서 제공하는 mtcars 데이터셋에서 mpg 와 다른 변수들 간의 상관 계수를 구하시오.
어느 변수가 mpg 와 가장 상관성이 높은지 산점도와 함께 제시하시오.

```
cor(mtcars$mpg,mtcars[-1])  
plot(mtcars$mpg~mtcars$drat,  
      data=mtcars,  
      las=1)
```

```
> cor(mtcars$mpg,mtcars[-1])  
      cyl      disp      hp      drat  
[1,] -0.852162 -0.8475514 -0.7761684 0.6811719  
      wt      qsec      vs      am      gear  
[1,] -0.8676594 0.418684 0.6640389 0.5998324 0.4802848  
      carb  
[1,] -0.5509251
```



Chapter 16. 연습문제 #3

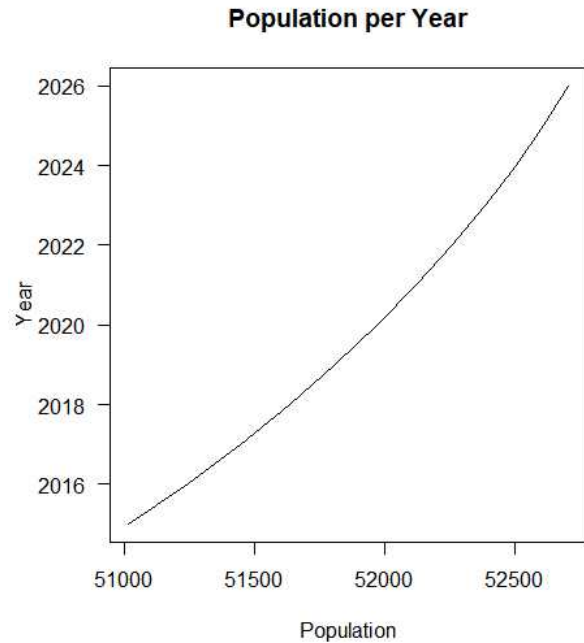
#3-1

1. 다음은 2015년부터 2026년도까지의 예상 인구수 추계자료이다. 선 그래프를 작성하시오.

연도	2015	2016	2017	2018	2019	2020	2021	2022	2023	2024	2025	2026
총인구(천명)	51,014	51,245	51,446	51,635	51,811	51,973	52,123	52,261	52,388	52,504	52,609	52,704

(출처: 국가통계포털 KOSIS)

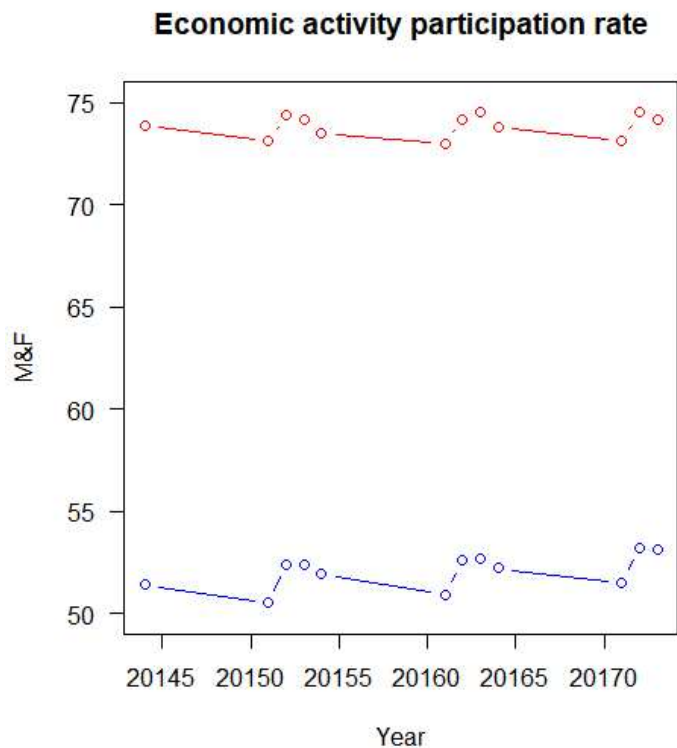
```
year <- 2015:2026
pop <- c(51014,51245,51446,51635,51811,51973,
         52123,52261,52388,52504,52609,52704)
plot(pop,
     year,
     main="Population per Year",
     type="l",
     lty=1,
     lwd=1,
     las=1,
     xlab="Population",
     ylab="Year")
```



#3-2

2. 다음은 2014년 4분기부터 2017년 3분기까지 남, 녀의 경제활동참가율 통계이다. 선 그래프를 작성하시오 (남,녀를 각각 다른 선으로 표시)(2014년 4분기는 20144, 2015년1분기는 20151 과 같이 입력한다

```
year <- c(20144,20151:20154,20161:20164,20171:20173)
m <- c(73.9,73.1,74.4,74.2,73.5,73.0,
       74.2,74.5,73.8,73.1,74.5,74.2)
f <- c(51.4,50.5,52.4,52.4,51.9,50.9,
       52.6,52.7,52.2,51.5,53.2,53.1)
plot(year,
     m,
     main="Economic activity participation rate",
     type="b",
     lty=1,
     col="red",
     xlab = "Year",
     ylab = "M&F",
     ylim = c(50,75),
     las=1)
lines(year,
     f,
     type="b",
     col="blue")
```



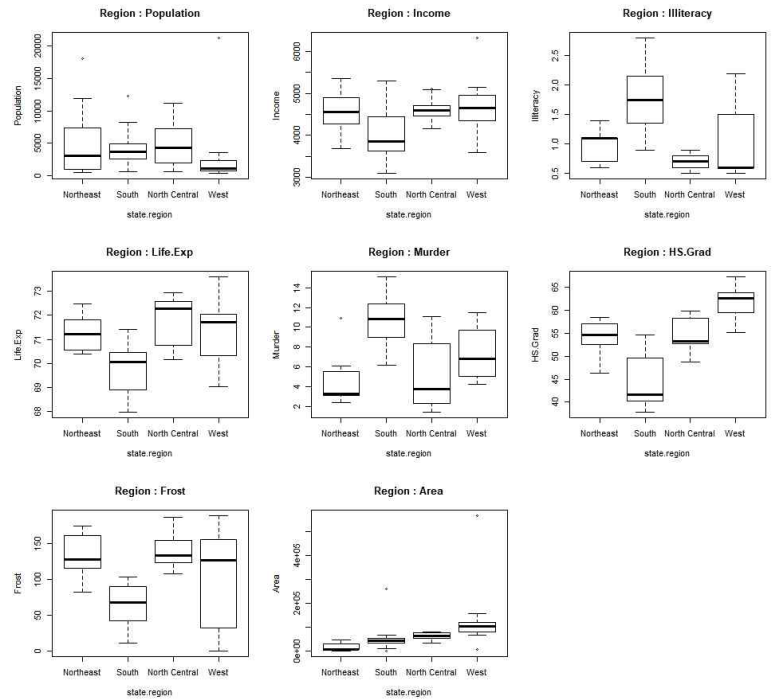
Chapter 16. 연습문제 #4

#4

- R 에서 제공하는 state.x77 (미국 50개주에 대한 통계)데이터셋을 분석해보시오
- State.x77 에 지역정보(state.region) 추가하여 분석

state.x77

```
st <- data.frame(state.x77, state.region)
st
par(mfrow=c(3,3))
boxplot(Population~state.region, data = st,
        main = "Region : Population")
boxplot(Income~state.region, data = st,
        main = "Region : Income")
boxplot(Illiteracy~state.region, data = st,
        main = "Region : Illiteracy")
boxplot(Life.Exp~state.region, data = st,
        main = "Region : Life.Exp")
boxplot(Murder~state.region, data = st,
        main = "Region : Murder")
boxplot(HS.Grad~state.region, data = st,
        main = "Region : HS.Grad")
boxplot(Frost~state.region, data = st,
        main = "Region : Frost")
boxplot(Area~state.region, data = st,
        main = "Region : Area")
```



Chapter 16. 연습문제 #5

#5-1

1. Emp 테이블에서 연봉을 1500 이상 받는 직원들의 모든 정보를 가져와 emp.high 데이터프레임에 저장하시오. emp.high의 내용을 보이시오

```
library(RMySQL)
mydb <- dbConnect(MySQL(), user='root',
                  password='1234', host='localhost',
                  dbname='my_db')
result <- dbSendQuery(mydb, "select * from emp where sal > 1500")
emp.high <- fetch(result, n = -1)
emp.high
dbClearResult(result)
```

```
> emp.high
  EMPNO ENAME      JOB  MGR  HIREDATE  SAL  COMM
1  7499 ALLEN  SALESMAN 7698 1981-02-20 1600  300
2  7566 PARK    MANAGER 7839 1981-04-02 2975   NA
3  7698 BLAKE    MANAGER 7839 1981-05-01 2850   NA
4  7782 CLARK    MANAGER 7839 1981-06-09 2450   NA
5  7788 SCOTT    ANALYST 7566 1982-12-09 3000   NA
6  7839 KING  PRESIDENT   NA 1981-11-17 5000   NA
7  7902 FORD     ANALYST 7566 1981-12-03 3000   NA
  DEPTNO
1      30
2      20
3      30
4      10
5      20
6      10
7      20
```

#5-2

2. 모든 직원의 직원번호, 이름, 급여, 입사일, 부서명, 부서위치를 emp.info 데이터프레임에 저장하시오. emp.info 의 내용을 보이시오

```
library(RMySQL)
mydb <- dbConnect(MySQL(), user='root',
                  password='1234', host='localhost',
                  dbname='my_db')
result <- dbSendQuery(mydb,
                    "select ename, sal, hiredate, dname, loc
                    from emp e, dept d
                    where e.deptno = d.deptno")
emp.info <- fetch(result, n = -1)
emp.info
dbClearResult(result)
```

```
> emp.info
  ename  sal  hiredate      dname      loc
1  CLARK 2450 1981-06-09 ACCOUNTING NEW YORK
2   KING 5000 1981-11-17 ACCOUNTING NEW YORK
3 MILLER 1300 1982-01-23 ACCOUNTING NEW YORK
4  SMITH  800 1980-12-17  RESEARCH  DALLAS
5   PARK 2975 1981-04-02  RESEARCH  DALLAS
6  SCOTT 3000 1982-12-09  RESEARCH  DALLAS
7 ADAMS 1100 1983-01-12  RESEARCH  DALLAS
8   FORD 3000 1981-12-03  RESEARCH  DALLAS
9  ALLEN 1600 1981-02-20    SALES  CHICAGO
10  WARD 1250 1981-02-22    SALES  CHICAGO
11 MARTIN 1250 1981-08-28    SALES  CHICAGO
12  BLAKE 2850 1981-05-01    SALES  CHICAGO
13 TURNER 1500 1981-08-08    SALES  CHICAGO
14  JAMES  950 1981-12-03    SALES  CHICAGO
```


#5-3

3. Dept 테이블에 아래와 같이 2개의 부서를 추가하시오.
Workbench 를 통해 2개의 부서가 추가되었는지 확인하시오.

```
library(RMySQL)
mydb <- dbConnect(MySQL(), user='root',
                  password='1234', host='localhost',
                  dbname ='my_db')
result <- dbGetQuery(mydb,
                    "insert into dept values
                    (70, 'Management', 'Paris'),
                    (80, 'Production', 'London')")
```

	DEPTNO	DNAME	LOC
▶	10	ACCOUNTING	NEW YORK
	20	RESEARCH	DALLAS
	30	SALES	CHICAGO
	40	OPERATIONS	BOSTON
	60	STRATEGY	SEOUL
	70	Management	Paris
	80	Production	London
	90	CLEANING	SEOUL
*	NULL	NULL	NULL

#5-4

4. emp 테이블의 모든 정보를 가져와 emp 데이터프레임에 저장하시오.
emp 데이터프레임에 대해 다음 작업을 R 로 수행하고 그 결과를 보이시오

```
library(RMySQL)
mydb <- dbConnect(MySQL(), user='root',
                  password='1234', host='localhost',
                  dbname ='my_db')
result <- dbSendQuery (mydb, "select * from emp")
emp <- fetch(result , n= -1)
emp
```

#5-4-(1)

(1) 모든 사원의 이름, 입사일자를 보이시오

```
emp[,c("ENAME","HIREDATE")]
```

```
> emp[,c("ENAME","HIREDATE")]
  ENAME  HIREDATE
1 SMITH 1980-12-17
2 ALLEN 1981-02-20
3 WARD  1981-02-22
4 PARK  1981-04-02
5 MARTIN 1981-08-28
6 BLAKE  1981-05-01
7 CLARK  1981-06-09
8 SCOTT  1982-12-09
9 KING   1981-11-17
10 TURNER 1981-08-08
11 ADAMS  1983-01-12
12 JAMES  1981-12-03
13 FORD   1981-12-03
14 MILLER 1982-01-23
```

#5-4-(2)

(2) 모든 사원의 급여합계를 보이시오

```
sum(emp[, "SAL"])
```

```
> sum(emp[, "SAL"])
[1] 29025
```


#5-4-(3)

(3) 모든 사원의 급여를 10% 올리려면 얼마의 비용이 추가로 필요한지 보이시오

```
sal <- emp[, "SAL"]
sal
salX1.1 <- sal * 1.1
sum(salX1.1) - sum(sal)
```

```
> sum(salX1.1) - sum(sal)
[1] 2902.5
```

#5-4-(4)

(4) 모든 사원의 이름, 급여, 10%올린 급여를 보이시오(힌트. cbind이용)

```
sal <- emp[, c("ENAME", "SAL")]
sal
salX1.1 <- sal*1.1
salX1.1
cbind(sal, salX1.1)
```

```
> cbind(sal, salX1.1)
      ENAME  SAL  salX1.1
1  SMITH    800    880.0
2  ALLEN   1600   1760.0
3   WARD   1250   1375.0
4   PARK   2975   3272.5
5 MARTIN   1250   1375.0
6  BLAKE   2850   3135.0
7  CLARK   2450   2695.0
8  SCOTT   3000   3300.0
9   KING   5000   5500.0
10 TURNER   1500   1650.0
11 ADAMS   1100   1210.0
12 JAMES    950   1045.0
13  FORD   3000   3300.0
14 MILLER   1300   1430.0
```

#5-4-(5)

(5) 급여를 2000 이상 받는 사원의 이름, 급여, 담당업무를 보이시오

```
emp1 <- emp[, c("ENAME", "SAL", "JOB")]
subset(emp1, SAL >= 2000)
```

```
> subset(emp1, SAL >= 2000)
      ENAME  SAL      JOB
4   PARK   2975  MANAGER
6  BLAKE   2850  MANAGER
7  CLARK   2450  MANAGER
8  SCOTT   3000  ANALYST
9   KING   5000 PRESIDENT
13  FORD   3000  ANALYST
```

#5-4-(6)

(6) 급여를 2000 이상 받고 부서번호가 20 인 사원의 이름, 급여, 입사일자를 보이시오

```
emp2 <- subset(emp, SAL >= 2000 & DEPTNO == 20)
emp2[, c("ENAME", "SAL", "HIREDATE")]
```

```
> emp2[, c("ENAME", "SAL", "HIREDATE")]
      ENAME  SAL  HIREDATE
4   PARK   2975 1981-04-02
8  SCOTT   3000 1982-12-09
13  FORD   3000 1981-12-03
```