



# 비즈니스프로그래밍

- Python과 R 기초 -



## 4. 데이터 분석



- ▶ 4.1 데이터 형태
  - ▶ R에 내장된 데이터 세트 보기
    - ▶ 데이터 목록 보기
    - ▶ 데이터 세트 정보 확인하기
    - ▶ 데이터 불러오기
  - ▶ 데이터 파악하기
    - ▶ 데이터의 앞부분 보기
    - ▶ 데이터의 뒷부분 보기
    - ▶ 데이터 세트 변수명 보기
    - ▶ 데이터 세트 데이터 내부구조 보기
    - ▶ 데이터 세트 데이터 차원 보기
    - ▶ 데이터 세트 요약 보기
  - ▶ 조회한 데이터 세트의 내용을 파일로 저장하고 저장된 파일 읽기



# R에 내장된 데이터



- ▶ 데이터 목록 보기

```
library(help = datasets)
```

- ▶ 데이터 세트 정보 확인하기

```
? 데이터세트이름
```

- ▶ 데이터 불러오기

```
data("데이터세트이름")
```



# 데이터 파악하기



- ▶ 데이터 앞부분 보기

```
head(데이터이름, n = 행의 수)
```

- ▶ 데이터 뒷부분 보기

```
tail(데이터이름, n = 행의 수)
```

- ▶ 데이터 세트 변수명 보기

```
names(데이터이름)
```

- ▶ 데이터 세트 데이터 내부구조 보기

```
str(데이터이름)
```

- ▶ 데이터 세트 데이터 차원 보기

```
dim(데이터이름)
```

- ▶ 데이터 세트 기초통계량 요약 보기

```
summary(데이터이름)
```

# 조회한 데이터 세트의 내용을 파일로 저장하고 저장된 파일 읽기

## ▶ 데이터 세트를 텍스트 파일로 저장하기

```
write.table(데이터프레임이름, "저장할경로/저장할파일이름",  
            [sep = " "/"",], [quote = FALSE/TRUE],  
            [row.names = FALSE/TRUE], ...)
```

## ▶ 텍스트 파일 읽기

```
데이터프레임이름 <- read.csv("저장된경로/저장된파일이름",  
                              header = T)
```



## 4. 데이터 분석



- ▶ 4.2 변수명 변경
  - ▶ names( ) 함수
  - ▶ reshape 패키지의 rename( ) 함수
  - ▶ dplyr 패키지의 rename( ) 함수



# 변수명 변경



- ▶ names( ) 함수

```
names(데이터이름) <- c("새변수명1", "새변수명2", "새변수명3", ...)
```

- ▶ reshape 패키지의 rename( ) 함수

```
rename(데이터세트, 기존변수명 = 새변수명)
```

- ▶ dplyr 패키지의 rename( ) 함수

```
rename(데이터세트, 새변수명 = 기존변수명)
```



## 4. 데이터 분석



- ▶ 4.3 파생 변수 생성
  - ▶ 파생 변수 란?
  - ▶ 기존 변수를 조합하여 파생 변수 만들기
  - ▶ 조건문을 활용하여 파생 변수 만들기
  - ▶ 중첩 조건문을 활용하여 파생 변수 만들기





# 파생 변수



- ▶ 파생 변수(Derived Variable): 기존의 변수를 연산 등을 통해 변형하여 만든 변수

이름	국어점수	영어점수	수학점수
aaa	70	85	75
bbb	60	90	80
ccc	90	80	85



이름	국어점수	영어점수	수학점수	총점
aaa	70	85	75	160
bbb	60	90	80	170
ccc	90	80	85	165

파생변수



# 파생 변수 생성



- ▶ 기존 변수를 조합하여 파생 변수 만들기 ➔ 연산 사용
- ▶ 조건문을 활용하여 파생 변수 만들기 ➔ ifelse( ) 함수 사용
- ▶ 중첩 조건문을 활용하여 파생 변수 만들기 ➔ ifelse( ) 함수를 여러 겹으로



## 5. 데이터 전처리




- ▶ 5.1 데이터 추출하기
  - ▶ 조건에 맞는 데이터만 추출하는 filter 함수
  - ▶ 필요한 변수를 추출하는 select 함수
  - ▶ dplyr 함수 조합하기

dplyr 함수	기능
filter( )	행 추출: 조건에 맞는 관측 값만 추출
select( )	열 추출: 필요한 변수만 추출

# 조건에 맞는 데이터 추출

- ▶ dplyr 패키지의 filter() 함수는 데이터 세트에서 조건에 맞는 행을 추출하고자 할 때 사용

name	department	worktime
aaa	관리부	45
bbb	영업부	48
ccc	홍보부	50
ddd	영업부	34
eee	영업부	44
fff	인사부	40
ggg	개발부	39



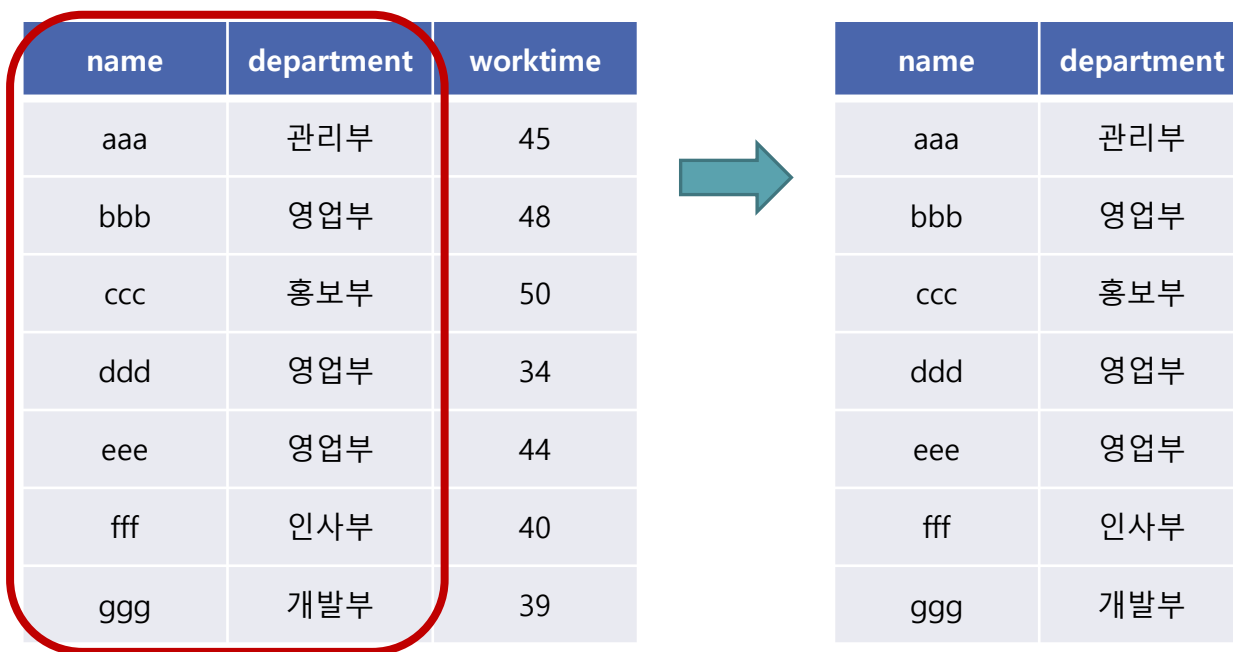
name	department	worktime
bbb	영업부	48
ddd	영업부	34
eee	영업부	44



# 필요한 변수 추출



- ▶ dplyr 패키지의 select( ) 함수는 데이터 세트에 포함되어 있는 여러 변수 중에서 필요한 변수만 추출하여 활용하고자 할 때 사용

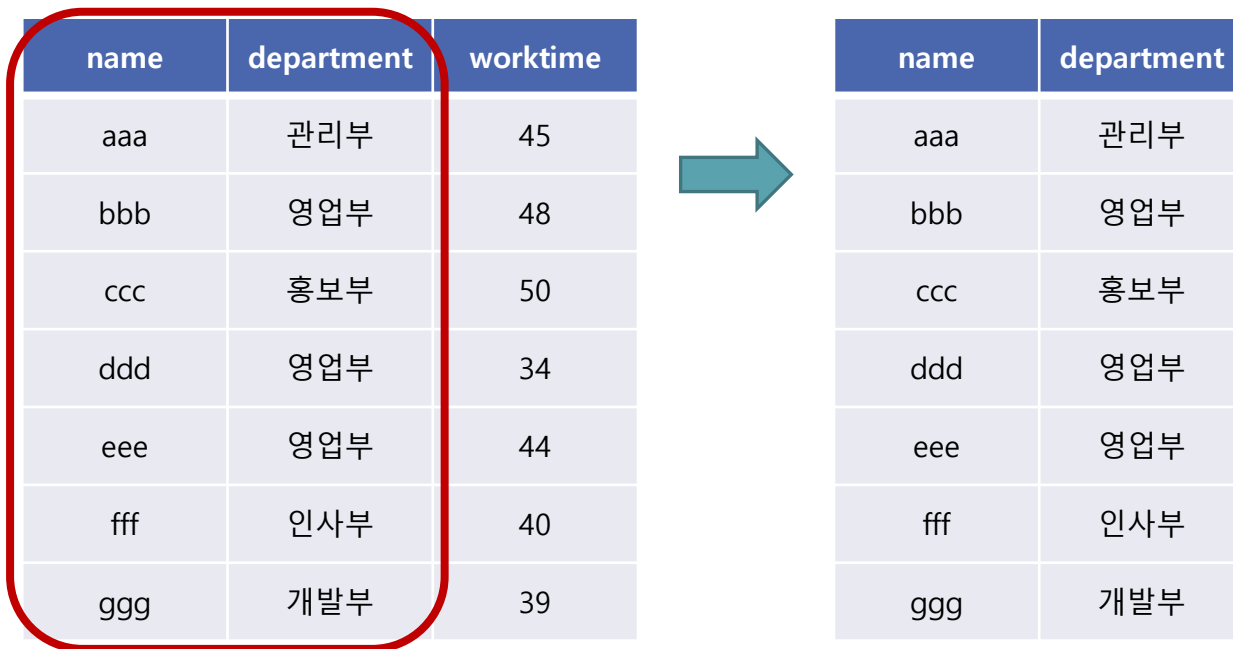


name	department	worktime
aaa	관리부	45
bbb	영업부	48
ccc	홍보부	50
ddd	영업부	34
eee	영업부	44
fff	인사부	40
ggg	개발부	39

name	department
aaa	관리부
bbb	영업부
ccc	홍보부
ddd	영업부
eee	영업부
fff	인사부
ggg	개발부

# dplyr 함수 조합하기

- ▶ dplyr 패키지의 함수를 조합할 때에는 %>% 연산자를 이용하여 조합하여 사용할 수 있다.



name	department	worktime
aaa	관리부	45
bbb	영업부	48
ccc	홍보부	50
ddd	영업부	34
eee	영업부	44
fff	인사부	40
ggg	개발부	39

name	department
aaa	관리부
bbb	영업부
ccc	홍보부
ddd	영업부
eee	영업부
fff	인사부
ggg	개발부



## 5. 데이터 전처리



- ▶ 5.2 데이터 정렬하기
  - ▶ 하나의 기준으로 정렬하기
  - ▶ 여러 개의 정렬 기준 적용하기
  - ▶ 추출과 정렬 조합하기

dplyr 함수	기능
arrange( )	정렬: 지정한 변수를 오름차순 or 내림차순 정렬



# 데이터 정렬하기



- ▶ `arrange()` 함수는 데이터를 오름차순 또는 내림차순으로 정렬할 때 사용
- ▶ 내림차순으로 정렬 ➔ `desc()` 를 함께 사용