05 데이터프레임

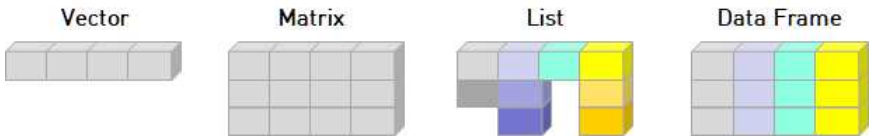
학습목표

1. 데이터프레임의 특징 및 변수와 관측치의 관계를 이해한다.
2. 다양한 방법으로 데이터프레임을 생성하고 변수에 이름을 부여할 수 있다.
3. 데이터프레임의 원소에 접근할 수 있다.
4. 데이터프레임에서 사용가능한 함수를 설명할 수 있다.

강의내용

R 자료구조 – 단일값들로 구성된 자료의 모음
– 벡터(vector), 행렬(matrix), 리스트(list), 데이터프레임(dataframe) 등

벡터(vector)	◦ 하나 이상의 원소로 이루어진 1차원 구조, R에서 가장 기본이 되는 자료구조 ◦ 동일한 자료형으로 구성되어야 함
행렬(matrix)	◦ 행과 열로 구성된 2차원 벡터
리스트(list)	◦ 다양한 자료형을 가질 수 있는 자료구조, 벡터의 확장형
데이터프레임 (dataframe)	◦ 데이터 분석에서 가장 많이 사용하는 테이블 형태의 2차원 자료구조 ◦ 각 열마다 다른 자료형을 가질 수 있으나 하나의 열은 동일한 자료형으로 구성



데이터프레임의 특징

– 테이블 형태의 2차원 구조
– 테이블의 각 열은 데이터의 특성을 나타내는 속성으로 변수(variable)이라고 하고,
각 행은 관측된 값 하나 하나를 나타내는 것으로 관측치(observation)이라고 함.
– 변수는 데이터 분석의 대상으로, 각 변수는 벡터나 팩터 형태의 자료이며, 모든 변수의 크기는 동일해야 한다.

변수 (Variable)

	사번	부서	성명	직위	연령	연봉
관측치 (Observation)	10201	경리부	백영진	과장	41	5,000만원
	40202	영업부	전혜진	부장	48	6,400만원
	30203	생산부	이용성	대리	44	4,700만원
	20204	마케팅부	김민지	과장	40	5,200만원
	30205	생산부	임동규	사원	27	3,200만원

테이블 형식으로 표현된 사원 정보

데이터프레임(dataframe) 생성

data.frame() 함수 이용	– 원하는 값을 입력하여 데이터프레임 생성 – 기본적으로 문자형을 factor형으로 만듦 문자형으로 만들려면 stringsAsFactors = F 옵션 사용한다.
외부 데이터파일 읽어오기	– read.csv() 함수를 이용하여 csv 형식의 자료를 읽어 데이터프레임 생성
R제공 Dataset	– R에서 제공하는 데이터셋 (패키지에 포함된 데이터셋)

```
# data.frame() 함수 사용
vd1 <- c("영수", "영미", "철수", "철이", "미애")
vd2 <- c(15, 14, 16, 13, 15)
vd3 <- c(T, F, T, T, F)
```

05 데이터프레임

```
student <- data.frame(name=vd1, age=vd2, sex=vd3)
student
str(student)     # obs 관측치(행)가 5개,  variables 변수(열)이 3개
                 # 이름이 문자형이 아닌 Factor 형

student <- data.frame(name=vd1, age=vd2, sex=vd3, stringsAsFactors = F)
str(student)      # 구조정보 다시 확인 : 이름이 문자형

# 외부 데이터 읽어오기
exam <- read.csv("csv_exam.csv")
exam

# R에서 제공하는 데이터셋 이용
mtcars
str(mtcars)

head(exam)       # 자료의 앞쪽 일부 확인
tail(exam)       # 자료의 뒷쪽 일부 확인
```

데이터프레임의 변수(속성)에 이름 붙이기

- data.frame() 함수에서 이름을 가진 데이터프레임 생성
- names()를 이용하여 변수에 이름을 붙일 수 있음
- dplyr 라이브러리인 rename() 함수를 이용하여 변수의 이름을 바꿀 수 있다.

  rename(dataframe, new1=old1, new2=old2, ...)

데이터프레임 내 원소에 접근

- 리스트와 동일한 방법으로, 열의 색인이나 이름을 이용하여 원소나 개별원소에 접근
- 행렬과 동일한 방법으로 행과 열의 색인이나 이름을 이용하여 원소에 접근
- 변수를 사용하기 때문에 주로 $ 기호를 사용한다.

```
# 데이터프레임 내 원소에 접근
student[3,1]
student[,c(1,2)]
student[,-1]
student[,c("sex","name")]

student[["name"]]
student$name              # 주로 데이터프레임에서 변수의 값을 읽어오는 방법
```

데이터프레임에서 유용한 함수

```
class(dataframe)      # 자료형 확인
str(dataframe)        # 자료의 구조 정보 확인
dim(dataframe)        # 자료의 차원 정보 (행, 열)
head(dataframe)       # 자료 첫 6개 행 확인
tail(dataframe)       # 자료 마지막 6개 행 확인
ncol(dataframe)       # 열(컬럼) 갯수
nrow(dataframe)       # 행 갯수
names(dataframe)      # 변수(컬럼) 이름
```