3. { Data 전처리(3) }

(R Study group 자료)

+) 지난 내용 요약

filter : 원하는 값(조건에 부합하는 값)을 추려냄
select : 원하는 변수만 선택

%>% : 파이프 연산자 “and than” 의 개념
코드를 더 간결하게 사용 가능

%in% : 조건들 중 하나라도 부합되면 (or의 개념)

1) 특정 기준으로 요약하기

특정한 기준으로 데이터를 요약해서 보고 싶은 경우가 대부분.
ex) titanic data에서 생존자(특정기준)가 몇 명인지(요약) ?

```
titanic  %>%
    group_by(Survived) %>%  # Survived 기준으로
    summarise(freq = n())  # 빈도로 요약해달라
                           # n() 은 빈도계산
```

기본구조

```
data %>%
    group_by(변수, 변수, …) %>%
    summarise(변수명 = function(),
              변수명 = function(), …)
```

1) 특정 기준으로 요약하기

select, filter 함수들을 활용하면 훨씬 더 효과적이다 !

titanic 데이터에서 Name, Survived, Embarked 만 추려내고,
그 중 생존자(Survived 가 1)의 수를 구하시요.

```
titanic %>%
        select(Name, Survived, Embarked) %>%
        group_by(Embarked) %>%
        filter(Survived == 1) %>%
        summarise(n = n())
```

1) 특정 기준으로 요약하기

select, filter 함수들을 활용하면 훨씬 더 효과적이다 !

titanic 데이터에서 Name, Survived, Pclass 만 추려내고,
그 중 생존자(Survived 가 1)의 수를 구하시요.

```
titanic %>%
        select(Name, Survived, Pclass) %>%
        group_by(Pclass) %>%
        filter(Survived == 1) %>%
        summarise(n = n())
```

1) 특정 기준으로 요약하기

select, filter 함수들을 활용하면 훨씬 더 효과적이다 !
특정 조건은 여러 개 가능하다.

titanic 데이터에서 Name, Survived, Pclass 만 추려내고,
그 중 생존 여부에 따라 빈도를 구하시요.

```
titanic %>%
        select(Name, Survived, Pclass) %>%
        group_by(Pclass, Survived) %>%
        summarise(n = n())
```

2) lol_data를 가지고 실습해보기

준비하기

```
library(tidyverse)

lol_df <- read.csv("lol_data.csv",
                   stringsAsFactor = F,
                   encoding = "euc-kr") # encoding = " " 인코딩설정
head(lol_df)
summary(lol_df)
str(lol_df)              # 자료형태 및 구조 확인하기
```

2) lol_data를 가지고 실습해보기

Q. 승/패의 빈도는 ?

```
lol_df  %>%
      group_by(win_loss) %>%
      summarise(freq = n()) %>%
      arrange(desc(freq))     # arrange() : 정렬 (오름차순)
                              # arrange(desc()) : 내림차순
```

기본구조

```
arrange(변수) # 변수를 기준으로 정렬해줌
              # defaul는 오름차순
              # desc() 내림차순 (= - 변수)
```

2) lol_data를 가지고 실습해보기

Q. 승/패에 따라 K/D/A, CS 차이가 있을까 ?

```
lol_df %>%
    group_by(win_loss) %>%
    summarise(mean_kill = mean(kill),
                      mean_dead = mean(dead),
                      mean_asist = mean(asist))
                      mean_cs = mean(cs)

# summarise != summary
# 여러 개의 요약 값을 만들 수 있다.
```

2) lol_data를 가지고 실습해보기

Q. 랭크, 자유랭크 게임 중 챔피언별 평균 평점이 높은 챔피언을 나열하시오.

```
lol_df %>%
    filter(game_type %in% c("자유랭크", "랭크")) %>%
    group_by(champ) %>%
    summarise(value = mean(mean_value)) %>%
    arrange(desc(value))
```

2) lol_data를 가지고 실습해보기

Q. 챔피언별 승점은 ? (승 : +1, 패 : -1)

```
lol_df %>%
    group_by(champ) %>%
    mutate(win_loss_nm = ifelse(win_loss == "승", 1, -1)) %>%
    summarise(sum = sum(win_loss_nm)) %>%
    arrange(sum)
```

기본구조

```
mutate(이름 = 계산식) # 새로이 변수를 생성해줌
ifelse(변수 조건, 참일 경우 값, 거짓일 경우 값)
```

2) lol_data를 가지고 실습해보기

+. 궁금증을 가지고 데이터를 바라보기 !!