



4. 제어문(Control)

chap04_Control 수업내용

- 조건문
 - ✓ if(), ifelse(), which()
- 반복문
 - ✓ for(), while()



4. 제어문

1) if()함수

```
if(x*y > 40){  
    cat("x*y의 결과는 40 이상입니다.\n") # \n 줄바꿈  
    cat("x*y =", z)  
}else{  
    cat("x*y의 결과는 40 미만입니다. x*y =", z, "\n")  
}
```



4. 제어문

2) `ifelse(조건, 참, 거짓)` - 3항 연산자 기능

`ifelse(score>=80, "우수","노력")` #우수

`ifelse(score<=80, "우수","노력")` #노력



4. 제어문

3) which 문

형식) `which()`의 괄호내의 조건에 해당하는 위치(인덱스)를 출력한다.

벡터에서 사용

```
name <- c("kim","lee","choi","park")
```

```
which(name=="choi") # [1] 3
```



4. 제어문

데이터프레임에서 사용

```
no <- c(1:5)
```

```
name <- c("홍길동", "이순신", "강감찬", "유관순", "김유신")
```

```
score <- c(85, 78, 89, 90, 74)
```

```
exam <- data.frame(학번=no, 이름=name, 성적=score)
```

```
exam
```

```
which(exam$이름=="유관순") # [1] 4, 없으면 0
```

```
exam[4,] # 4번째 레코드 보기
```



4. 제어문

1) 반복문

형식) **for(변수 in 값) {표현식} - 단일문{} 생략 가능**

```
i <- c(1:10)
```

```
for(n in i){ # 10회 반복
```

```
  print(n * 10) # 계산식(numeric만 가능) 출력
```

```
  print(n)
```

```
}
```



4. 제어문

```
for(n in i){  
  if(n%%2==0){  
    next # 다음문장 skip -> 반복문 계속  
  }else{  
    print(n) # 홀수만 출력  
  }  
}
```

```
[1] 1  
[1] 3  
[1] 5  
[1] 7  
[1] 9
```



4. 제어문

```
# 데이터 파일의 변수명 출력  
name <- c(names(exam))  
for(n in name){ # 변수명 출력  
  print(n)  
}
```

```
[1] "학번"  
[1] "이름"  
[1] "성적"
```




4. 제어문

```
score = c(85, 95, 98)
```

```
name = c('홍길동', '이순신', '강감찬')
```

```
i <- 1
```

```
for (s in score){
```

```
  cat(name[i], " -> ", s, "\n")
```

```
  i <- i + 1
```

```
}
```

```
홍길동 -> 85
```

```
이순신 -> 95
```

```
강감찬 -> 98
```



4. 제어문

2) 반복문 - while(조건){표현식}

```
i = 0
while(i < 10){
  i <- i + 1
  print(i) # 1~10까지 출력됨
}
```