PR6

PR6\_201821479\_황혜린

2018년 10월 28일

# 함수와 사용자정의 함수

## 함수

* 특정 목적에 맞게 생성된 연산과정의 집합
* ex)mean함수 : 모든 원소의 합을 원소의 개수로 나눔

## 사용자 정의 함수

* 사용자의 편의에 따라 직접 작성하여 사용하는 함수
* 함수명 = function(인수){연산과정}형태로 작성(한가지 연산만 할 경우 {}로 묶지 않아도 됨)
* 연산과정으로 나오는 결과값을 return, print, cat등으로 변환하는 형태가 이상적

### 예시1. 두 숫자를 비교해 더 큰 수를 반환하는 함수

compare <- function(x,y) if(x>y) cat(x) else cat(y)  
compare(10,20)

## 20

### 예시2. 평균값과 표준오차를 계산하는 함수

se <- function(x) {  
 tmp.sd <- sd(x)  
 tmp.N <- length(x)  
 tmp.se <- tmp.sd / sqrt(tmp.N)  
 return(tmp.se)  
}  
  
A<-c(1,2,3,4,5,6,7,8,9,10)  
se(A)

## [1] 0.9574271

### 예시3. 데이터 프레임의 앞 뒤 3개의 데이터를 리스트로 보여주는 함수

head\_tail <- function(x){  
 front <- head(x, 3)  
 rear <- tail(x, 3)  
 F\_R <- list(front, rear)  
 return(F\_R)  
}  
  
head\_tail(mtcars)

## [[1]]  
## mpg cyl disp hp drat wt qsec vs am gear carb  
## Mazda RX4 21.0 6 160 110 3.90 2.620 16.46 0 1 4 4  
## Mazda RX4 Wag 21.0 6 160 110 3.90 2.875 17.02 0 1 4 4  
## Datsun 710 22.8 4 108 93 3.85 2.320 18.61 1 1 4 1  
##   
## [[2]]  
## mpg cyl disp hp drat wt qsec vs am gear carb  
## Ferrari Dino 19.7 6 145 175 3.62 2.77 15.5 0 1 5 6  
## Maserati Bora 15.0 8 301 335 3.54 3.57 14.6 0 1 5 8  
## Volvo 142E 21.4 4 121 109 4.11 2.78 18.6 1 1 4 2

### 예시4. 홀수 판별 함수

oddnum <- function(x){  
 if(x%%2==1){  
 return(T)  
 } else {  
 return(F)  
 }  
}

### scope of variable

* 함수 바깥에서 생성된 변수는 같은 함수 안에서는 언제나 사용가능
* 함수 안에서 생성된 변수는 함수가 종료되면 사라짐
* 함수 내에서 생성된 변수가 사라지지 않게 하려면 “<<-”을 할당연산자로 사용

scopetest <- function(x){  
 a <- 10   
 print(a)  
 print(x)  
}  
  
scopetest(9)

## [1] 10  
## [1] 9

#print(a)   
  
scopetest <- function(x){  
 a <<- 10  
 print(a)  
 print(x)  
}  
  
scopetest(9)

## [1] 10  
## [1] 9

print(a)

## [1] 10

# 연습문제

in.rate <- function(x,y){  
 revenue <- ((x - y) / y) \* 100  
 return(revenue)  
}  
a <- c(11905000, 11973000, 12190000, 12700000, 12303000, 12604000) #금일의 종가   
b <- c(0, 11905000, 11973000, 12190000, 12700000, 12303000) #첫날의 전일의 종가는 알 수 없기 때문에 0으로 적었습니다  
in.rate(a,b)

## [1] Inf 0.5711886 1.8124113 4.1837572 -3.1259843 2.4465578