

# Base R

## Cheat Sheet

### 사용법 도움받기

#### 매뉴얼 보기

?mean

특정 함수에 대한 도움 받기

help.search('weighted mean')

특정 단어나 문구에 대한 사용법 찾기

help(package = 'dplyr')

패키지 사용법 찾기

#### 오브젝트 정보 더보기

str(iris)

오브젝트의 구조에 대한 요약 보기

class(iris)

오브젝트가 소속된 클래스 찾기

### 패키지 사용하기

install.packages('dplyr')

CRAN에서 패키지를 다운로드하고 설치

library(dplyr)

세션에 패키지 불러오기

dplyr::select

패키지에서 특정 함수 사용

data(iris)

내장 데이터셋을 environment에 불러오기

### 작업 디렉토리

getwd()

현재 작업 디렉토리 찾기

setwd('C://file/path')

현재 작업 디렉토리 변경

RStudio의 project를 사용해 작업 디렉토리를 현재 폴더로 변경할 수 있음

### 벡터

#### 벡터 생성하기

c(2, 4, 6)	2 4 6	주어진 원소를 가지는 벡터 생성
2:6	2 3 4 5 6	정수 수열
seq(2, 3, by=0.5)	2.0 2.5 3.0	복잡한 수열
rep(1:2, times=3)	1 2 1 2 1 2	벡터 반복
rep(1:2, each=3)	1 1 1 2 2 2	벡터의 원소 반복

#### 벡터 함수

sort(x)                      rev(x)  
x 정렬해서 반환              x 뒤집어서 반환

table(x)                      unique(x)  
값들의 개수 보기.              값들의 종류 보기.

#### 벡터 원소 선택

위치로 선택

x[4]                      네번째 원소  
x[-4]                      네번째 빼고 전부  
x[2:4]                      두번째부터 네번째까지  
x[-(2:4)]                      두번째부터 네번째까지 빼고 전부

x[c(1, 5)]                      첫번째와 다섯번째 원소

값으로 선택

x[x == 10]                      값이 10인 원소  
x[which(x==10)]                      값이 10인 원소  
x[x < 0]                      0보다 작은 모든 원소  
x[x%in%c(1, 2, 5)]                      {1, 2, 5}에 속한 원소

이름으로 선택

x['apple']                      이름이 'apple'인 원소.

### 프로그래밍

#### For문

```
for (variable in sequence) {  
  Do something  
}
```

예시

```
for (i in 1:4) {  
  j <- i + 10  
  print(j)  
}
```

#### If문

```
if (condition) {  
  Do something  
} else {  
  Do something  
}
```

예시

```
if (i > 3) {  
  print('Yes')  
} else {  
  print('No')  
}
```

#### While문

```
while (condition) {  
  Do something  
}
```

예시

```
while (i < 5) {  
  print(i)  
  i <- i + 1  
}
```

#### 함수

```
func_name <- function(var) {  
  Do something  
  return(new_variable)  
}
```

예시

```
square <- function(x) {  
  squared <- x*x  
  return(squared)  
}
```

### 데이터 입출력

#### readr 패키지 참조

입력	출력	설명
df <- read.table('file.txt')	write.table(df, 'file.txt')	구분 문자로 분리된 텍스트 파일 입출력
df <- read.csv('file.csv')	write.csv(df, 'file.csv')	csv 파일 입출력 (read.table/write.table의 특수한 경우)
load('file.RData')	save(df, file = 'file.RData')	R 데이터 파일 입출력

관계 연산자

a == b	같다	a > b	크다 (초과)	a >= b	크거나 같다 (이상)	is.na(a)	값이 없다
a != b	같지 않다	a < b	작다 (미만)	a <= b	작거나 같다 (이하)	is.null(a)	값이 null

## 자료형

자주 쓰는 자료형간의 변환. 표 아래쪽의 값에서 위쪽의 값으로는 항상 변환 가능하다.

as.logical	TRUE, FALSE, TRUE	Boolean 값 (참 아니면 거짓)
as.numeric	1, 0, 1	정수나 실수
as.character	'1', '0', '1'	문자열. 팩터에서 자주 사용함
as.factor	'1', '0', '1' levels: '1', '0'	level이 정해진 문자열. 몇몇 통계모델에 필요함

## 수학 함수

log(x)	자연로그	sum(x)	합
exp(x)	지수함수	mean(x)	평균
max(x)	최댓값	median(x)	중앙값
min(x)	최솟값	quantile(x)	사분위수
(x, n)	소수점 n째자리까 지 반올림	rank(x)	원소의 큰 순서
(x, n)	유효숫자 n자리	var(x)	분산
(x, y)	상관계수	sd(x)	표준편차.

## 변수 할당

```
> a <- 'apple'
> a
[1] 'apple'
```

## Environment

**ls()** environment의 모든 변수 나열  
**rm(x)** environment에서 x 제거  
**rm(list = ls())** environment에서 모든 변수 제거  
**RStudio의 environment 패널로 environment**  
 의 변수들을 볼 수 있음




## 리스트

```
l <- list(x = 1:5, y = c('a', 'b'))
```

리스트는 자료형에 상관없이 여러 개의 원소를 가지는 집합

<code>l[[2]]</code> l의 두번째 원소	<code>l[1]</code> 첫 번째 원소만 있는 새 리스트	<code>l\$x</code> x라는 이름의 원소	<code>l['y']</code> y라는 이름의 원소만 있는 새 리스트.
-------------------------------------	----------------------------------------------	------------------------------------	----------------------------------------------------

## 행렬

 <code>m[2, ]</code> - 열 선택	<code>t(m)</code> 전치
 <code>m[, 1]</code> - 행 선택	<code>m %*% n</code> 행렬곱
 <code>m[2, 3]</code> - 원소 선택	<code>solve(m, n)</code> <code>m*x = n</code> 의 해 찾기

## 데이터프레임

### dplyr 패키지 참조

```
df <- data.frame(x = 1:3, y = c('a', 'b', 'c'))
```

모든 원소의 길이가 같은 리스트의 특수한 케이스

x	y
1	a
2	b
3	c

### 리스트로 자르기

<code>df\$x</code>	<code>df[[2]]</code>
--------------------	----------------------



### 데이터프레임 파악하기

`view(df)` 전체 데이터프레임 보기  
`head(df)` 첫 6열까지만 보기

### 행렬로 자르기

<code>df[, 2]</code>	<code>nrow(df)</code> 열 개수
<code>df[2, ]</code>	<code>ncol(df)</code> 행 개수
<code>df[2, 2]</code>	<code>dim(df)</code> 행과 열의 개수

### cbind - 열로 합치기

	
<code>rbind - 행으로 합치기</code>	

## 문자열

### stringr 패키지 참조

<code>paste(x, y, sep = '')</code>	여러 개의 벡터 합치기
<code>paste(x, collapse = '')</code>	벡터의 원소들 합치기
<code>grep(pattern, x)</code>	x에서 정규 표현식으로 찾기
<code>gsub(pattern, replace, x)</code>	x에서 해당하는 항목 대체하기
<code>toupper(x)</code>	대문자화하기
<code>tolower(x)</code>	소문자화하기
<code>nchar(x)</code>	문자열 길이

## 팩터

<b>factor(x)</b> 벡터를 팩터로 바꾸기. 팩터의 level과 order 설정 가능	<b>cut(x, breaks = 4)</b> 수로 이루어진 벡터를 잘라서 팩터로 만들기
---------------------------------------------------------------	---------------------------------------------------------

## 통계




<b>lm(y ~ x, data=df)</b> 선형모형	<b>t.test(x, y)</b> t-검정	<b>prop.test</b> 비율검정
<b>glm(y ~ x, data=df)</b> 일반화 선형모형	<b>pairwise.t.test</b> 짝지어진 데이터에 대한 t-검정	<b>aov</b> 분산 분석 (ANOVA)
<b>summary</b> 모델에서 더 자세한 정보 불러오기.		

## 분포

	난수	확률밀도함수	누적분포함수	분위수
정규분포	<b>rnorm</b>	<b>dnorm</b>	<b>pnorm</b>	<b>qnorm</b>
푸아송 분포	<b>rpois</b>	<b>dpois</b>	<b>ppois</b>	<b>qpois</b>
이항분포	<b>rbinom</b>	<b>dbinom</b>	<b>pbinom</b>	<b>qbinom</b>
균등분포	<b>runif</b>	<b>dunif</b>	<b>punif</b>	<b>qunif</b>

## 플로팅

### ggplot2 패키지 참조

 <b>plot(x)</b> x의 값 (순서대로)	 <b>plot(x, y)</b> x에 대한 y의 값	 <b>hist(x)</b> x의 히스토그램.
------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------

## 날짜

### lubridate 패키지 참조