**<2DAYS>**

**# : 주석, 화면에 입력되지 않고 부연설명.**

**# ; : 한 줄에 여러 명령 가능.**

**# class() : 자료 확인 ex) numeric, matrix, data frame**

**# print() : 출력하기**

**# 사칙연산 : +, - , \* , / ,%%(나머지),%/%(몫),^(제곱)**

**# 산술 연산 함수**

**Log(), sqrt() – 제곱근, max(), min(), abs() – 절대값**

**Factorial, sin(pi/n), cos(pi/n), tan(pi/n)**

**# 변수(variable) : 메모리 영역에 대한 이름, 값을 저장하는 메모리 영역 (alt + -로 지정)**

**# str() : 변수의 내용(type, 값) 확인 함수**

**# NA : 결측치(missing value) # ‘10’ : 영숫자(문자로 된 숫자)**

**# is로 시작하는 함수 : 확인 함수(결과 논리값)**

**Ex) is.numeric(a) 🡪 TRUE or FALSE**

**# as로 시작하는 함수 : 형태 변환 함수**

**Ex) as.numeric(‘a’) 🡪 ‘a’를 숫자로 전환**

**# 벡터(Vector) : 동일 자료형 값의 집합, 1차원 배열**

**# c() : connect 의 약자, 벡터 생성 함수**

**# 1:10에서 콜론은 연속적 범위 지정 1~10**

**# seq(1,5,2) : sequence 순서 함수, 첫번째-시작, 두번째-끝,세번째-간격 ex) seq(1,5,2) -> 1,3,5**

**# rep(1, times=3) : repeat 반복 함수, 대상과 횟수**

**# names() : 벡터 원소 값에 이름 지정 함수**

**# length() : 전체 벡터의 길이 계산. 3개, 5개…..**

**# [ ] : 벡터 안의 요소 값을 구할 때 사용(요소 지정)**

**Ex) a,<- c(1,2,3) / a[2] = 2**

**# 벡터 요소 읽기**

**Ex) v<- c(1,2,3,4,5) v[-3] = 1,2,4,5 / -3을 입력하면 3번째 인수를 제외하고 읽음.**

**# 벡터에 적용 가능한 함수**

**sum() : 합계, mean() : 평균, median() : 중앙값**

**max() : 최대값, min() : 최소값, sort() : 정렬**

**sort(v, decreasing = FALSE) : v 값의 범위를 오름차순 정리**

**sort(v, decreasing = TRUE) : v 값의 범위를 내림차순 정리**

**range() : 값의 범위(최소값,최대값)**

**var() : 분산 sd() : 표준편차**

**# 관계, 논리 연산자**

**관계 연산자 : >, >=, <, <=, ==, !=(같지 않다)**

**논리 연산자 : &,&&(and), |,||(or), !(not)**

**<3DAYS>**

**# 벡터 이어서**

**All() : 모든 요소가 조건을 만족하는가? TRUE or FALSE 나옴**

**Any() : 요소 중 조건에 만족하는 것이 있는가? TRUE of FALSE**

**Head() : 처음부터 일정 개수 ( 초기값, default : 6개)**

**Tail() : 끝에서부터 일정 개수 ( 초기값, default : 6개)**

**# 집합(set) : 중복 데이터가 없는 벡터**

**Union(x,y) : 합집합 intersect(x,y) : 교집합**

**Setdiff(x,y) : 차집합 setequal(x,y) : 동일 요소 존재 확인**

**# List : 동일 자료형, 다른 자료형을 저장할 수 있는 벡터**

**List() : list생성 함수,요소는 key(이름) = value(값) 형식**

**Ex) v <- c( 90, 85, 70, 84 )**

**l <- list( name = 'hong', age = 30, status = TRUE,**

**score = v )**

**# factor() : 팩터형 벡터 생성 함수**

**bt <- c( 'A', 'B', 'B', 'O', 'AB', 'A' ) # 문자형 벡터**

**bt.factor <- factor( bt ) # 팩터형 벡터**

**# matrix() : matrix 생성 함수**

**Ex) z <- matrix( 1:20, nrow = 4, ncol = 5, byrow = T )**

**m1 <- cbind( x, y ) : 행(데이터)으로 묶음**

**m2 <- rbind( x, y ) : 열(변수)로 묶음**

**rownames( score ) <- c( 'hong', 'kim', 'lee', 'yoo' ) # 행이름**

**colnames( score ) <- c( 'english', 'math', 'science' ) # 열이름**

**data.frame() : data frame을 생성하는 함수**

**$ : 데이터의 문자에 접근할 때 사용 Ex) iris$Species**

**# matrix / dataframe 에서 사용하는 함수**

**dim() : 몇 행 몇 열**

**nrow() : 행의 개수**

**ncol() : 열의 개수**

**unique( iris[, 5 ] ) : 어떤 종류로 구성되어 있는지 이름**

**table( iris[ , 5 ] ) : 이름과 데이터 수**