

```
#####
### Computational Policy and Project Analysis #####
### Lecture title: Lecture 03. Data Sturcture #####
### Developed by. KKIM #####
#####
```

```
##### Vector #####
```

```
### Vector
c("Lee", "Yoon", "Shim", "Ahn", "Oh", "Huh")
```

```
PresCand <- c("Lee", "Yoon", "Shim", "Ahn", "Oh", "Huh")
PresCand
```

```
### Vector index
name_vector <- c("Christiano", "Leo", "Paul", "Wayne")
name_vector[1:3]
name_vector[-2]
name_vector[c(-1, -2)]
name_vector[c(1, 3, 4)]
```

```
### Vector Indexing
# vector with some missing values
sample_vector <- c(1, 4, NA, 2, 1, NA, 4, NA)
sample_vector[1:5]
sample_vector[c(1, 3, 5)]
sample_vector[-1]
sample_vector[c(-1, -3, -5)]
sample_vector[c(T, T, F, T, F, T, F, T)]
is.na(sample_vector)
sum(is.na(sample_vector))
```

```
### Vector with named elements I
named_vector <- c("JYP", "Dancer")
named_vector
names(named_vector) <- c("Name", "Occupation")
named_vector
```

```
named_vector['Name']
named_vector['Occupation']
named_vector[1]
```

```
### Vector with named elements II
weather_vector <- c("Mon" = "Sunny", "Tues" = "Rainy",
                  "Wed" = "Cloudy", "Thur" = "Foggy",
                  "Fri" = "Sunny", "Sat" = "Sunny",
                  "Sun" = "Cloudy")

weather_vector
names(weather_vector)
```

```
### Quiz
# 1) Fill the blank below
pizzaSize <- c("M", "L", "L", "XL", "XXL",
              "S", "M", "S", "S")
FpizzaSize <- factor(pizzaSize,
                    levels=c("S", "M", "L", "XL"))
str(FpizzaSize)
```

```
# 2) What is the expected result of summary(FpizzaSize)
summary(FpizzaSize)
summary(pizzaSize)
```

```
# 3) What is expected result of women[2:4,]?
data(women)
women
women[2:4,]
```

```
# 4) What is expected result of sum(is.na(women$height))?
sum(is.na(women$height))
```

```

### Short-cut for Making Numeric Vector
a_vector <- 1:10      ## numbers from 1 to 10
a_vector <- c(1,2,3,4,5,6,7,8,9,10)    ## numbers from 1 to 10
b_vector <- seq(1, 10, 2) ## numbers from 1 to 10 increasing by 2

a_vector
b_vector

c_vector <- rep(1:3, 3)
d_vector <- rep(1:3, each = 3)

c_vector
d_vector
c(a_vector, b_vector) ## combine vectors to single vector

### What happen when you combine two vectors with different data type?
numeric_vector <- c(1, 10, 49)
character_vector <- c("a", "b", "c")
c(character_vector, numeric_vector)
typeof(c(character_vector, numeric_vector))

### Vector-related functions
a_vector <- c(1,5,2,7,8, 2, 3)
b_vector <- seq(1, 10, 3)

a_vector
b_vector

intersect(a_vector, b_vector) ## intersection
union(a_vector, b_vector)     ## union
setdiff(a_vector, b_vector)   ## set difference
unique(a_vector)              ## find distinct members

### Basic Vector operations
a_vector <- c(1,5,2,7,8)
b_vector <- seq(1, 10, 2)

sum(a_vector) ## summation
mean(a_vector) ## average

# operation of Vector and Scala
a_vector + 10

a_vector > 4
sum(a_vector > 4) ## what does this mean?

# operation of Vector and Vector
a_vector - b_vector
a_vector == b_vector
sum(a_vector == b_vector) ## what does this mean?

##### Matrix #####

### Matrix
matrix(1:9)
matrix(1:10, nrow = 3)
matrix(1:9, byrow=FALSE, nrow = 3)
matrix(1:9, byrow=TRUE, nrow = 3)

### Naming a Matrix
# Box office Star Wars (in millions!)
new_hope <- c(460.998, 314.4)
empire_strikes <- c(290.475, 247.900)
return_jedi <- c(309.306, 165.8)

# Construct matrix
star_wars_matrix <-
  matrix(c(new_hope, empire_strikes, return_jedi),

```

```

      nrow = 3,
      byrow = TRUE)
star_wars_matrix

# Vectors region and titles, used for naming
region <- c("US", "non-US")
titles <- c("A New Hope",
            "The Empire Strikes Back",
            "Return of the Jedi")

# Name the columns with region
colnames(star_wars_matrix) <- region
# Name the rows with titles
rownames(star_wars_matrix) <- titles
star_wars_matrix

### Row-wise and Column-wise Summation
# The worldwide box office figures
rowSums(star_wars_matrix)
# Total revenue for entire Series
colSums(star_wars_matrix)
# The worldwide box office figures
rowMeans(star_wars_matrix)
# Total revenue for entire Series
colMeans(star_wars_matrix)

### Adding new column
# The worldwide box office figures
worldwide_vector <- rowSums(star_wars_matrix)
worldwide_vector

# Bind the new variable worldwide_vector as a column to star_wars_matrix
all_wars_matrix <-
  cbind(star_wars_matrix,
        worldwide_vector)
all_wars_matrix

### Adding new row
# Construct star_wars_matrix2
box_office <- c(474.5, 552.5, 310.7,
               338.7, 380.3, 468.5)
star_wars_matrix2 <-
  matrix(box_office,
        nrow = 3,
        byrow = TRUE,
        dimnames = list(c("The Phantom Menace",
                          "Attack of the Clones",
                          "Revenge of the Sith"),
                        c("US", "non-US")))

star_wars_matrix
star_wars_matrix2
all_wars_matrix <- rbind(star_wars_matrix,
                        star_wars_matrix2)
all_wars_matrix

### Selection of Matrix Elements
all_wars_matrix[1:3,1]
all_wars_matrix[1:3,'non-US']
all_wars_matrix[, 'US']
all_wars_matrix[c(1,3,5),]

### Matrix Computations
A.mat <- matrix(1:9, byrow = TRUE, nrow = 3)
B.mat <- matrix(rep(1:3,each = 3), byrow = TRUE, nrow = 3)
C.mat <- matrix(rep(1:3, 2), byrow = F, ncol = 2)

A.mat
B.mat
C.mat

```

```

## matrix operation with other matrix
## (element-wise operation)
A.mat * B.mat
A.mat - B.mat
A.mat / B.mat
A.mat %*% C.mat

##### DataFrame #####

### DataFrame
data(mtcars)
mtcars

### Creating DataFrame
# Definition of vectors
name <- c("Mercury", "Venus", "Earth", "Mars", "Jupiter", "Saturn", "Uranus", "Neptune")
type <- c("Terrestrial planet", "Terrestrial planet", "Terrestrial planet",
         "Terrestrial planet", "Gas giant", "Gas giant", "Gas giant", "Gas giant")
diameter <- c(0.382, 0.949, 1, 0.532, 11.209, 9.449, 4.007, 3.883)
rotation <- c(58.64, -243.02, 1, 1.03, 0.41, 0.43, -0.72, 0.67)
rings <- c(FALSE, FALSE, FALSE, FALSE, TRUE, TRUE, TRUE, TRUE)
# save(name, type, diameter, rotation, rings,
#       file="R file/R file_LEC03/planets_df_vectors.RData")

### Create a data frame from the vectors
# Method 1
# load(file="R file/R file_LEC02/planets_df_vectors.RData")
planets_df <-
  data.frame(name, type,
            diameter, rotation, rings)
planets_df

# Method 2
my.df <- data.frame(
  name = c('John', 'Kim', 'Kaith'),
  job = c('Teacher', 'Policeman', 'Secetary'),
  age = c(32, 25, 28))
my.df

### Selection of DataFrame Elements
# Print out diameter of Mercury (row 1, column 3)
planets_df[1,3]

# Print out data for Mars (entire fourth row)
planets_df[4, ]

# you can use of directly variable name
# Select first 5 values of diameter column
planets_df[1:5, 'diameter']

planets_df[,3]
planets_df[, "diameter"]
planets_df$diameter

# find planets with rings
planets_df$rings
planets_df[planets_df$rings, ]

# select names of planets with rings
planets_df[planets_df$rings, 'name']

# find planets with larger diameter than earth
planets_df$diameter > 1
planets_df[planets_df$diameter > 1, ]

### Quiz
mymat <- matrix(1:20, ncol=5)
mydf <- data.frame(mymat)

```

```

# 1) What are the expected results?
length(mymat)
length(mydf)
# sum(mymat)
# sum(mydf)

# 2) Compare the two versions
# left
mymat <- matrix(1:20, ncol=5)
mymat
mymat <- cbind(mymat, c("21","22","23","24"))
mymat
sum(mymat)
sum(mymat[,-6])

# write
mymat <- matrix(1:20, ncol=5)
mydf <- data.frame(mymat)
mydf <- cbind(mydf, X6=c("21","22","23","24"))

mydf
sum(mydf)
mydf$X1
mydf$X6
sum(mydf[,-6])
length(mydf)

##### List #####

### List
# Vector with numerics from 1 up to 10
my_vector <- 1:10
my_vector

# Matrix with numerics from 1 up to 9
my_matrix <- matrix(1:9, nrow = 3)
my_matrix

# First 10 elements of the built-in data frame mtcars
my_df <- mtcars[1:2,]
my_df

# Construct list with these different elements:
my_list <- list(my_vector,
               my_matrix,
               my_df)
my_list

### List
# Adapt list() call to give the components names
my_list <- list(vec = my_vector,
               mat = my_matrix,
               df = my_df)

# Print out my_list
my_list
names(my_list)
names(my_list) <- c("Vector",
                  "Matrix",
                  "DataFrame")
names(my_list)
my_list

### Selection of List Elements
my_list[[1]][1]
my_list[1]$Vector[1]
my_list[[3]]

```

```

my_list[['Vector']]
my_list['Vector']

my_list$Vector

### Adding components to the list
my_list$new_vector <- c(1,3,5,7,9)
str(my_list)

my_list[['new_vector']]

### Quiz
data(mtcars)

# 1) Use mtcars data set to create a list called
# 'Listlist' with following conditions
Listlist <- list(mtcars[mtcars$gear==5,],
                mtcars[mtcars$gear==4,],
                mtcars[mtcars$gear==3,])

# 2) Add names to the list, "FiveGear" , "FourGear" , "ThreeGear"
names(Listlist)<-c("FiveGear",
                  "FourGear",
                  "ThreeGear")

# 3) Add following vector as the fourth element of Listlist: "God" ," is" ," Good"
Listlist[[4]] <- c("Yonsei","is","Love")

# 4) Write down R code that returns 1st row of Listlist' s FourGear
# (Write down the two versions)
Listlist[[2]][1,]
Listlist[['FourGear']][1,]
Listlist$FourGear[1,]

head(Listlist)
lapply(Listlist, head)

```