List

Code

List란 (그 전에 리뷰와)

- **Vectors** (one dimensional array): can hold numeric, character or logical values. The elements in a vector all have the same data type.
- **Matrices** (two dimensional array): can hold numeric, character or logical values. The elements in a matrix all have the same data type.
- **Data frames** (two-dimensional objects): can hold numeric, character or logical values. Within a column all elements have the same data type, but different columns can be of different data type.
- **Lists** : A list in R is similar to your to-do list at work or school: the different items on that list most likely differ in length, characteristic, type of activity that has to do be done

–

리스트 만들기

Hide

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

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

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

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

my_list
```

–

List 이름 할당하기

Hide

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

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

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

# Adapt list() call to give the components names
my_list <- list(my_vector, my_matrix, my_df)
names(my_list) <- c("vec", "mat", "df")
# Print out my_list
my_list
```

–

List안으로 접근하기

문법	의미
x$name	리스트 x에서 키 값 name에 해당하는 값
x[n]	리스트 x에서 n번째 데이터와 서브리스트
x[[n]]	리스트 x에서 n번째 저장된 값

Hide

```
# shining_list is already pre-loaded in the workspace
x <- list(name = "foo", height = c(1, 3, 5))
# Print out the vector representing the actors
x[2]

# Print the second element of the vector representing the actors
x$height[3]
```

–

List 추가하기

Hide

```
# shining_list, the list containing movie name, actors and reviews, is pre-loaded in
 the workspace
x <- list(name = "foo", height = c(1, 3, 5))
# We forgot something; add the year to shining_list
x_plus <- c(x, year = 1980)
# Have a look at shining_list_full
str(x)
```