

Learning Outcomes

At the end of the session, you will be able to:

- Write, run, and explain the difference between vector and list

Activity

1. R Vector

1.1. Vector Construction

- Write and run the following in R Console. Make your conclusion about the code:

```
# Creating a sequence from 5 to 13.
v <- 5:13
print(v)
# Creating a sequence from 6.6 to 12.6.
v <- 6.6:12.6
print(v)
# If the final element specified does not belong to the
sequence then it is discarded.
v <- 3.8:11.4
print(v)
# Create vector with elements from 5 to 9 incrementing by 0.4.
print(seq(5, 9, by = 0.4))
# The logical and numeric values are converted to characters.
s <- c('apple','red',5,TRUE)
print(s)
```

1.2. Accessing Vector Elements

- Write and run the following in R Console. Make your conclusion about the code:

```
# Accessing vector elements using position.
t <- c("Sun","Mon","Tue","Wed","Thurs","Fri","Sat")
u <- t[c(2,3,6)]
print(u)
# Accessing vector elements using logical indexing.
v <- t[c(TRUE,FALSE,FALSE,FALSE,FALSE,TRUE,FALSE)]
print(v)
# Accessing vector elements using negative indexing.
x <- t[c(-2,-5)]
print(x)
# Accessing vector elements using 0/1 indexing.
y <- t[c(0,0,0,0,0,0,1)]
print(y)
```

1.3. Vector Manipulation

- Write and run the following in R Console. Make your conclusion about the code:

```
# Create two vectors.
v1 <- c(3,8,4,5,0,11)
v2 <- c(4,11,0,8,1,2)
# Vector addition.
add.result <- v1+v2
print(add.result)
```

```
# Vector subtraction.
sub.result <- v1-v2
print(sub.result)
# Vector multiplication.
multi.result <- v1*v2
print(multi.result)
# Vector division.
divi.result <- v1/v2
print(divi.result)
```

1.4. Vector Element Recycling

- Write and run the following in R Console. Make your conclusion about the code:

```
v1 <- c(3,8,4,5,0,11)
v2 <- c(4,11)
# V2 becomes c(4,11,4,11,4,11)
add.result <- v1+v2
print(add.result)
sub.result <- v1-v2
print(sub.result)
```

1.5. Vector Element Sorting

- Write and run the following in R Console. Make your conclusion about the code:

```
v <- c(3,8,4,5,0,11, -9, 304)
# Sort the elements of the vector.
sort.result <- sort(v)
print(sort.result)
# Sort the elements in the reverse order.
revsort.result <- sort(v, decreasing = TRUE)
print(revsort.result)
# Sorting character vectors.
v <- c("Red","Blue","yellow","violet")
sort.result <- sort(v)
print(sort.result)
# Sorting character vectors in reverse order.
revsort.result <- sort(v, decreasing = TRUE)
print(revsort.result)
```

2. R List

2.1. List Construction

- Write and run the following in R Console. Make your conclusion about the code:

```
# Create a list containing strings, numbers, vectors and a
logical
# values.
list_data <- list("Red", "Green", c(21,32,11), TRUE, 51.23,
119.1)
print(list_data)
```

2.2. Naming List Elements

- Write and run the following in R Console. Make your conclusion about the code:

```
# Create a list containing a vector and a list.
list_data <- list(c("Jan","Feb","Mar"), list("green",12.3))
# Give names to the elements in the list.
names(list_data) <- c("1st_Quarter", "A_Inner_list")
# Show the list.
print(list_data)
```

2.3. Accessing List Elements

- Write and run the following in R Console. Make your conclusion about the code:

```
# Create a list containing a vector and a list.
list_data <- list(c("Jan","Feb","Mar"), list("green",12.3))
# Give names to the elements in the list.
names(list_data) <- c("1st_Quarter", "A_Inner_list")
# Access the first element of the list.
print(list_data[1])
# Access the thrid element. As it is also a list, all its
elements will be printed.
print(list_data[2])
# Access the list element using the name of the element.
print(list_data$A_Inner_list)
# Get index of list element using value.
print(which(list_data$`1st_Quarter` == "Feb"))
```

2.4. List Manipulation

- Write and run the following in R Console. Make your conclusion about the code:

```
# Create a list containing a vector and a list.
list_data <- list(c("Jan","Feb","Mar"), list("green",12.3))
# Give names to the elements in the list.
names(list_data) <- c("1st_Quarter", "A_Inner_list")
# Add element at the end of the list.
list_data[3] <- "New_element"
print(list_data[3])
# Remove the last element.
list_data[3] <- NULL
# Print the 3rd Element.
print(list_data[3])
# Update the 2nd Element.
list_data[2] <- "updated_element"
print(list_data[2])
```