Lecture note on R break and next statements:

## **R break statement**

The break statement is used to terminate the execution of a loop. This will stop any further iterations.

The syntax of the break statement is:

Code snippet

if (test\_expression) {

break

}

The break statement is often used inside a conditional (if...else) statement in a loop. If the condition inside the test\_expression returns True, then the break statement is executed.

For example,

Code snippet

# vector to be iterated over

x = c(1, 2, 3, 4, 5, 6, 7)

# for loop with break statement

for(i in x) {

# if condition with break

if(i == 4) {

break

}

print(i)

}

Output:

Code snippet

[1] 1

[2] 2

[3] 3

Here, we have defined a vector of numbers from 1 to 7. Inside the for loop, we check if the current number is 4 using an if statement.

If yes, then the break statement is executed and no further iterations are carried out. Hence, only numbers from 1 to 3 are printed.

## break statement in nested loop

If you have a nested loop and the break statement is inside the inner loop, then the execution of only the inner loop will be terminated.

Let's check out a program to use break statements in a nested loop.

Code snippet

# vector to be iterated over

x = c(1, 2, 3)

y = c(1, 2, 3)

# nested for loop with break statement

for(i in x) {

for (j in y) {

if (i == 2 & j == 2) {

break

}

print(paste(i, j))

}

}

Output:

Code snippet

[1] "1 1"

[1] "1 2"

[1] "1 3"

[1] "2 1"

[1] "3 1"

[1] "3 2"

[1] "3 3"

Here, we have a break statement inside the inner loop.

We have used it inside a conditional statement such that if both the numbers are equal to 2, the inner loop gets terminated.

The flow then moves to the outer loop. Hence, the combination (2, 2) is never printed.

## **R next statement**

In R, the next statement skips the current iteration of the loop and starts the loop from the next iteration.

The syntax of the next statement is:

Code snippet

if (test\_condition) {

next

}

If the program encounters the next statement, any further execution of code from the current iteration is skipped, and the next iteration begins.

Let's check out a program to print only even numbers from a vector of numbers.

Code snippet

# vector to be iterated over

x = c(1, 2, 3, 4, 5, 6, 7, 8)

# for loop with next statement

for(i in x) {

# if condition with next

if(i %% 2 != 0) {

next

}

print(i)

}

Output:

Code snippet

[1] 2

[1] 4

[1] 6

[1] 8

Here, we have used an if statement to check whether the current number in the loop is odd or not.

If yes, the next statement inside the if block is executed, and the current iteration is skipped.

I hope this helps! Let me know if you have any other questions.

XXXXXXX—-------------XXXXXXXX

Sure, here is a lecture note on R break and next statements:

**Break Statement**

The break statement can be used to terminate a loop. The syntax of the break statement is as follows:

Code snippet

break

The break statement can be used anywhere inside a loop body. When the break statement is executed, the loop terminates immediately.

Example

The following code prints the numbers from 1 to 10:

Code snippet

for (i in 1:10) {

print(i)

}

The following code uses a break statement to terminate the loop when the value of i is equal to 5:

Code snippet

for (i in 1:10) {

if (i == 5) {

break

}

print(i)

}

**Next Statement**

The next statement can be used to skip an iteration of a loop. The syntax of the next statement is as follows:

Code snippet

next

The next statement can be used anywhere inside a loop body. When the next statement is executed, the current iteration of the loop is skipped and the next iteration begins.

Example

The following code prints the even numbers from 1 to 10:

Code snippet

for (i in 1:10) {

if (i %% 2 != 0) {

next

}

print(i)

}

This code will print the following output:

Code snippet

[1] 2

[1] 4

[1] 6

[1] 8

[1] 10

As you can see, the code skips the odd numbers and prints only the even numbers.

XXXXXXX—------XXXXXXX

Sure, here's a lecture note on R break and next:

The break and next statements are used to alter the flow of a program. These are also known as jump statements in programming:

* break - terminate a looping statement
* next - skips an iteration of the loop

R break Statement

You can use a break statement inside a loop (for, while, repeat) to terminate the execution of the loop. This will stop any further iterations.

The syntax of the break statement is:

Code snippet

if (test\_expression) {

break

}

The break statement is often used inside a conditional (if...else) statement in a loop. If the condition inside the test\_expression returns True, then the break statement is executed. For example,

Code snippet

# vector to be iterated over

x = c(1, 2, 3, 4, 5, 6, 7)

# for loop with break statement

for(i in x) {

# if condition with break

if(i == 4) {

break

}

print(i)

}

Output

Code snippet

[1] 1

[1] 2

[1] 3

Here, we have defined a vector of numbers from 1 to 7. Inside the for loop, we check if the current number is 4 using an if statement.

If yes, then the break statement is executed and no further iterations are carried out. Hence, only numbers from 1 to 3 are printed.

break Statement in Nested Loop

If you have a nested loop and the break statement is inside the inner loop, then the execution of only the inner loop will be terminated.

Let's check out a program to use break statements in a nested loop.

Code snippet

# vector to be iterated over

x = c(1, 2, 3)

y = c(1, 2, 3)

# nested for loop with break statement

for(i in x) {

for (j in y) {

if (i == 2 & j == 2) {

break

}

print(paste(i, j))

}

}

Output

Code snippet

[1] "1 1"

[1] "1 2"

[1] "1 3"

[1] "2 1"

[1] "3 1"

[1] "3 2"

[1] "3 3"

Here, we have a break statement inside the inner loop.

We have used it inside a conditional statement such that if both the numbers are equal to 2, the inner loop gets terminated.

The flow then moves to the outer loop. Hence, the combination (2, 2) is never printed.

R next Statement

In R, the next statement skips the current iteration of the loop and starts the loop from the next iteration.

The syntax of the next statement is:

Code snippet

if (test\_condition) {

next

}

If the program encounters the next statement, any further execution of code from the current iteration is skipped, and the next iteration begins.

Let's check out a program to print only even numbers from a vector of numbers.

Code snippet

# vector to be iterated over

x = c(1, 2, 3, 4, 5, 6, 7, 8)

# for loop with next statement

for(i in x) {

# if condition with next

if(i %% 2 != 0) {

next

}

print(i)

}

Output

Code snippet

[1] 2

[1] 4

[1] 6

[1] 8

Here, we have used an if statement to check whether the current number in the loop is odd or not.

If yes, the next statement inside the if block is executed, and the current iteration is skipped.