## R While Loop

A while loop is a control flow statement that allows code to be executed repeatedly as long as a certain condition is true. The syntax of a while loop in R is as follows:

Code snippet

while (condition) {

# code to be executed repeatedly

}

The condition is evaluated before each iteration of the loop. If the condition is true, the code inside the loop body is executed. If the condition is false, the loop terminates.

## Example

The following code prints the numbers from 1 to 10:

Code snippet

# initialize a counter

counter <- 1

# while loop to print numbers from 1 to 10

while (counter <= 10) {

print(counter)

counter <- counter + 1

}

## Break Statement

The break statement can be used to terminate a loop early. 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.

## Next Statement

The next statement can be used to skip the current 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 is started.

## Exercises

1. Write a while loop that prints the even numbers from 2 to 10.
2. Write a while loop that prints the Fibonacci sequence up to the 10th term.
3. Write a while loop that asks the user for a number and prints "Too high!" if the number is greater than 100, "Too low!" if the number is less than 100, and "Correct!" if the number is equal to 100.

Xxxxx—----------xxx

What is a while loop?

A while loop is a control flow statement that allows code to be executed repeatedly as long as a given boolean condition is true.

Syntax of a while loop:

Code snippet

while (condition) {

# code to be executed repeatedly

}

Where is a while loop used?

While loops are used in many different contexts, but they are commonly used to:

* Repeat a block of code until a certain condition is met.
* Execute a block of code a certain number of times.
* Process a list or vector of data one element at a time.

Example of a while loop:

Code snippet

# Print the numbers from 1 to 10

number <- 1

while (number <= 10) {

print(number)

number <- number + 1

}

Output:

Code snippet

1

2

3

4

5

6

7

8

9

10

How to break out of a while loop:

The break statement can be used to break out of a while loop. The break statement will cause the while loop to terminate immediately, even if the condition is still true.

Example of breaking out of a while loop:

Code snippet

# Print the numbers from 1 to 10, but break out of the loop if the number is 5

number <- 1

while (number <= 10) {

print(number)

if (number == 5) {

break

}

number <- number + 1

}

Output:

Code snippet

1

2

3

4

How to continue to the next iteration of a while loop:

The next statement can be used to continue to the next iteration of a while loop. The next statement will cause the current iteration of the loop to be skipped, and the next iteration will be executed instead.

Example of continuing to the next iteration of a while loop:

Code snippet

# Print the even numbers from 1 to 10

number <- 1

while (number <= 10) {

if (number %% 2 == 1) {

next

}

print(number)

number <- number + 1

}

Output:

Code snippet

2

4

6

8

10

Exercises:

1. Write a while loop that prints the numbers from 1 to 100.
2. Write a while loop that asks the user for a number and keeps asking until the user enters a number that is greater than 10.
3. Write a while loop that reads a list of numbers from the user and prints the sum of the numbers.

Sure, here is a lecture note on R while loop:

What is a while loop?

A while loop is a control flow statement that allows code to be executed repeatedly as long as a specified condition is true. The syntax of a while loop is as follows:

Code snippet

while (condition) {

# code to be executed repeatedly

}

The condition is evaluated before each iteration of the loop. If the condition is true, the code inside the loop body is executed. If the condition is false, the loop terminates.

Example of a while loop

The following code prints the numbers from 1 to 10:

Code snippet

# initialize a variable

i <- 1

# while loop to print numbers from 1 to 10

while (i <= 10) {

print(i)

i <- i + 1

}

Break statement

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

Code snippet

break

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

Next statement

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

Code snippet

next

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

Exercises

1. Write a while loop that prints the even numbers from 2 to 10.
2. Write a while loop that prints the prime numbers from 2 to 100.
3. Write a while loop that asks the user for a number and prints "Too low" if the number is less than 10, "Too high" if the number is greater than 10, and "Just right" if the number is equal to 10.