

# Control Structures

**PRESENTED BY**

*Vinod Raju*  
*Data Science Practitioner*

## TOC

[Introduction Control Structures](#)

[Different Control Structures](#)

[Summary](#)

Topic Number	Topic Name
1	<a href="#">Introduction to Control Structures</a>
2	<a href="#">Different Control Structures</a>
3	<a href="#">Summary</a>

## TOC

[Introduction Control Structures](#)

[Different Control Structures](#)

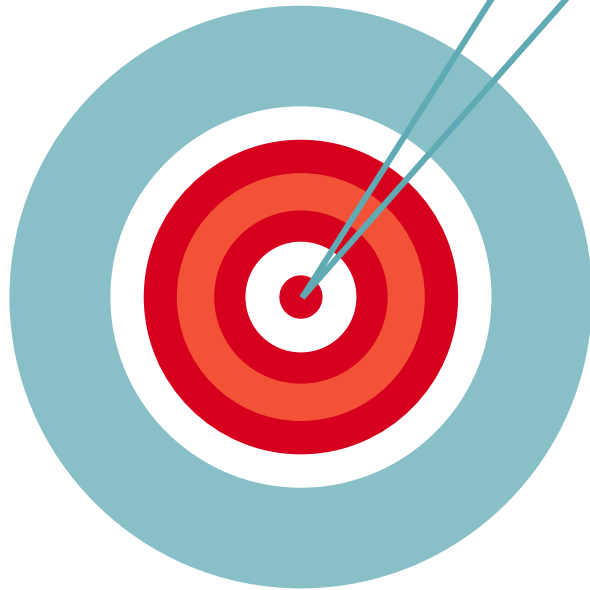
[Summary](#)

## Learning Objectives

By the end of this unit, you will be able to:

— Explain different control structures

— Apply control structures in R programming



## TOC

[Introduction Control Structures](#)

[Different Control Structures](#)

[Summary](#)

# Introduction to Control Structures

## TOC

[Introduction Control Structures](#)

[Different Control Structures](#)

[Summary](#)

# What are Control Structures?

Control structures allow you to control the flow of execution of a program. A few common control structures are:

if, else

for

while

repeat

break

next

return

## TOC

[Introduction Control Structures](#)

[Different Control Structures](#)

[Summary](#)

# Different Control Structures

## TOC

## If –Else condition

[Introduction Control Structures](#)[Different Control Structures](#)[Summary](#)

On the condition holding TRUE, the code block following the 'if' statement gets executed; on the failure of the condition, the 'else' block gets executed.

### Pseudocode:

```
if (condition) {  
    # do something  
} else {  
    # do something else}
```

### Example:

```
x <- 5  
if(x > 0){  
    print("Non-negative number")  
} else {  
    print("Negative number")}
```

## TOC

[Introduction Control Structures](#)[Different Control Structures](#)[Summary](#)

# Loops (for loops)

A 'for' loop works on an iterable variable picking up values, typically from a vector and assigns successive values till the end of the collection is reached.

### Example:

```
for (i in 1:3) { # i in 1:3 can also be in the form of (i in c(1,2,3))  
  print(i)  
}
```

### Output:

1

2

3



## TOC

[Introduction Control Structures](#)

[Different Control Structures](#)

[Summary](#)

# While Loop

The 'while' loops are used to loop until a specific condition is met.

The test expression is evaluated and the body of the loop is entered, only if the result is TRUE.

The statements inside the loop are executed and the flow returns to evaluate the test expression again.

This is repeated until the test expression evaluates to FALSE, in which case, the loop exits.

## TOC

[Introduction Control Structures](#)[Different Control Structures](#)[Summary](#)

# While Loop

When the statements inside the loop are executed, the flow returns to evaluate the test expression again.

### Pseudocode:

```
while (test_expression)  
{  
    statement  
}
```

### Example:

```
i=0  
while(i<10)  
{print(i)  
  i=i+1}
```

## TOC

[Introduction Control Structures](#)[Different Control Structures](#)[Summary](#)

# Repeat and Break

‘Repeat and break’ executes a sequence of statements multiple times and abbreviates the code that manages the loop variable.

## Pseudocode:

```
repeat {  
    commands  
    if(condition) {  
        break  
    }  
}
```

## Example:

```
i=0  
repeat {  
    print(i)  
    if(i==10) {break} # Break statement is used to exit the loop  
    i=i+1 }
```

## TOC

[Introduction Control Structures](#)[Different Control Structures](#)[Summary](#)

## Next

A 'next' statement is used to skip the current iteration of a loop without terminating it. On encountering 'next', the R parser skips further evaluation and starts next iteration of the loop.

### Example:

```
x <- 1:5
for (val in x) {
  if (val == 3){
    next
  }
  print(val)}
```

## TOC

[Introduction Control Structures](#)[Different Control Structures](#)[Summary](#)

## Return

Return statement is used to explicitly return a desired value from a function.

### **Pseudocode:**

*return(expression)*

### **Example:**

```
mysum = function(a,b)
{
  s = a+b
  return(s)
}
print(mysum(10,20))
```

## TOC

[Introduction Control Structures](#)

[Different Control Structures](#)

[Summary](#)

# Summary

## TOC

[Introduction Control Structures](#)

[Different Control Structures](#)

[Summary](#)

## Summary

In this unit, you learnt:

- Control structures allow to control the flow of execution of a program.
- The different control structures are:
  - if, else
  - for
  - while
  - repeat
  - break
  - next
  - return

**THANK  
YOU!**

Copyright Manipal Global Education Services Pvt. Ltd. All Rights Reserved.

*All product and company names used or referred to in this work are trademarks or registered trademarks of their respective holders. Use of them in this work does not imply any affiliation with or endorsement by them.*

*This work contains a variety of copyrighted material. Some of this is the intellectual property of Manipal Global Education, some material is owned by others which is clearly indicated, and other material may be in the public domain. Except for material which is unambiguously and unarguably in the public domain, permission is not given for any commercial use or sale of this work or any portion or component hereof. No part of this work (except as legally allowed for private use and study) may be reproduced, adapted, or further disseminated without the express and written permission of Manipal Global Education or the legal holder of copyright, as the case may be.*