Flow Control in R: if-else, while, for, switch, break, and next

Introduction

Flow control structures allow you to execute code conditionally or repeatedly, based on logical tests. This section covers the main flow control constructs in R: `if-else`, `while`, `for`, `switch`, `break`, and `next`.

1. if-else

- Use `if` to execute code if a condition is `TRUE`.
- Use `else` to specify alternative actions if the condition is `FALSE`.

Example:

```
x <- 5
if (x > 0) {
    print("Positive")
} else {
    print("Non-positive")
}
```

2. while

- Use `while` to execute a block of code repeatedly while a condition is `TRUE`.

Example:

```
i <- 1
while (i <= 5) {
    print(i)
    i <- i + 1
}
```

3. for

- Use `for` to iterate over elements in a vector or sequence.

Example:

```
for (i in 1:5) {
    print(i)
}
```

4. switch

- Use `switch` to execute one of several options based on a key.

Example:

```
choice <- "B"
result <- switch(choice,
                 "A" = "Option A selected",
                 "B" = "Option B selected",
                 "C" = "Option C selected")
print(result)
```

5. break

- Use `break` to exit a loop prematurely.

Example:

```
for (i in 1:10) {
    if (i == 5) {
        break
    }
    print(i)
}
```

6. next

- Use `next` to skip the current iteration of a loop and continue with the next one.

Example:

```
for (i in 1:5) {
    if (i == 3) {
        next
    }
    print(i)
}
```

Exercises

1. Write an `if-else` statement to check if a number is even or odd.
2. Create a script using `if-else` to classify a number as positive, negative, or zero.
3. Use a `while` loop to calculate the sum of integers from 1 to 10.
4. Write a `while` loop to print all even numbers less than 20.
5. Create a `for` loop to iterate over a vector of names and print each name.
6. Write a `for` loop to calculate the factorial of a number.
7. Use `switch` to print the day of the week based on an input number (1 for Monday, 2 for Tuesday, etc.).
8. Write a `switch` statement to return the season based on the input month.
9. Use a `for` loop with `break` to find the first number divisible by 7 in a sequence.
10. Write a `for` loop with `next` to print all numbers from 1 to 10 except multiples of 3.
11. Use `if-else` inside a `for` loop to print whether each number in a vector is positive or negative.
12. Write a `while` loop to generate the Fibonacci sequence up to 100.
13. Create a nested `for` loop to print a multiplication table for numbers 1 to 5.
14. Write a `for` loop with `break` to stop iterating when the sum of the numbers exceeds 50.
15. Use `switch` to assign grades based on a numeric score: 90-100 (A), 80-89 (B), etc.

Conclusion

These exercises cover conditional statements, loops, and flow control mechanisms in R. Mastering these concepts will enhance your ability to write efficient and logical R scripts.