lab02-operators-loops

Joseph Davis Chamdani

2025-10-07

1 Basic R

1.1 Convert degrees F to degrees C

(a) Assign the temperature today (in Fahrenheit)

```
fahrenheit <- 58
fahrenheit
```

```
## [1] 58
```

(b) Compute the temperature in Celsius

```
celsius <- (5/9) * (fahrenheit - 32)
celsius
```

```
## [1] 14.44444
```

(c) Show the results using inline code

Today's temperature is **58°F** and in celcius it's **14.44°C**.

1.2 Work with strings

(a) Create a variable for your hometown

```
hometown <- "Jakarta 🇮🇩"
hometown
```

```
## [1] "Jakarta 🇮🇩"
```

(b) Create a variable for your name

```
name <- "Joseph Davis Chamdani"
name
```

```
## [1] "Joseph Davis Chamdani"
```

(c) Combine both using `paste()`

```
message <- paste("My name is", name, "and I'm from", hometown)
message
```

```
## [1] "My name is Joseph Davis Chamdani and I'm from Jakarta 🇮🇩"
```

The result from the message is:
My name is Joseph Davis Chamdani and I'm from Jakarta 🇮🇩

1.3 Computations

(a) Define the variables

```
m <- 70
p <- 100
w <- "sunny"
m; p; w
```

```
## [1] 70
```

```
## [1] 100
```

```
## [1] "sunny"
```

(b) Is money more than 100?

```
m > 100
```

```
## [1] FALSE
```

(c) Is money greater than or equal to price?

```
m >= p
```

```
## [1] FALSE
```

(d) Is it a sad day?

A day is sad if price > money *and* weather is not sunny.

```
sad_day <- (p > m) & (w != "sunny")
sad_day
```

```
## [1] FALSE
```

(e) Repeat with money set to 100

```
m <- 100

# (3b) Is money > 100?
m > 100
```

```
## [1] FALSE
```

```
# (3c) Is money >= price?
m >= p
```

```
## [1] TRUE
```

```
# (3d) Sad day condition
sad_day <- (p > m) & (w != "sunny")
sad_day
```

```
## [1] FALSE
```

2 Asivärk

2.1 Loop

Print “o” n times in the same row

```
n <- 8

for (i in 1:n) {
  cat("o")
}
```

```
## oooooooo
```

2.2 “Asivärk” pattern

Write a for-loop that creates “asivärk”

```
n <- 6

for (i in 1:n) {
  for (j in 1:i) {
    cat("o")
  }
  cat("\n")
}
```

```
## o
## oo
## ooo
## oooo
## ooooo
## oooooo
```

2.3 Inverted “Asivärk”

Write a loop that creates inverted “asivärk”

```
n <- 6

for (i in n:1) {
  for (j in 1:i) {
    cat("|")
  }
  cat("\n")
```

```
}
```

```
## ||||||
## |||||
## ||||
## |||
## ||
## |
```

2.4 Combine inverted and normal “Asivärk”

Write a loop that combines the inverted and the normal “asivärk”

```
n <- 7

for (i in 1:n) {
  for (j in i:n) {
    cat("|")
  }
  for (k in 1:i) {
    cat("o")
  }
  cat("\n")
}
```

```
## |||||||o
## ||||||oo
## |||||ooo
## ||||oooo
## |||ooooo
## ||oooooo
## |ooooooo
```

3 if/else

3.1 Print numbers from 1 to 10

Use a for-loop to print numbers 1:10

```
for (i in 1:10) {
  cat(i, ": \n")
}
```

```
## 1 :
## 2 :
## 3 :
## 4 :
## 5 :
## 6 :
## 7 :
## 8 :
## 9 :
## 10 :
```

3.2 Print “even” after each even number

Used the modulo operator `%%` to check evenness

```
for (i in 1:10) {
  if (i %% 2 == 0) {
    cat(i, ": even\n")
  } else {
    cat(i, ": \n")
  }
}
```

```
## 1 :
## 2 : even
## 3 :
## 4 : even
## 5 :
## 6 : even
## 7 :
## 8 : even
## 9 :
## 10 : even
```

3.3 Print “even” or “odd” for each number

Use if/else to print a label after every number

```
for (i in 1:10) {
  if (i %% 2 == 0) {
```

```
    cat(i, ": even\n")
  } else {
    cat(i, ": odd\n")
  }
}
```

```
## 1 : odd
## 2 : even
## 3 : odd
## 4 : even
## 5 : odd
## 6 : even
## 7 : odd
## 8 : even
## 9 : odd
## 10 : even
```

4 Accumulating loops

4.1 Sum of numbers 1 + 2 + 3 + ... + 10

Use a for-loop to calculate the total sum

```
sum <- 0

for (i in 1:10) {
  sum <- sum + i
}

sum
```

```
## [1] 55
```

4.2 Factorial 1 × 2 × 3 × ... × 10

Multiply through a for-loop

```
factorial <- 1

for (i in 1:10) {
  factorial <- factorial * i
}

factorial
```

```
## [1] 3628800
```

4.3 Product of 1 × (1/2) × (1/3) × ... × (1/10)

```
product <- 1

for (i in 1:10) {
  product <- product * (1 / i)
}

product
```

```
## [1] 2.755732e-07
```