

PRAKTIKUM 3

FUNGSI DALAM R

Tujuan Umum Praktikum

Mahasiswa diharapkan mampu memahami konsep dasar fungsi dalam R, serta mampu mengimplementasikannya dalam berbagai skenario pemrograman, seperti pengolahan data, analisis statistik, dan pembuatan kode yang modular dan efisien.

Tujuan Khusus

1. Mahasiswa mampu mendefinisikan dan memanggil fungsi dasar dalam R.
2. Mahasiswa mampu menggunakan argumen fungsi, baik argumen posisi maupun argumen dengan nama (*named arguments*).
3. Mahasiswa mampu membuat fungsi dengan beberapa parameter input dan mengembalikan hasil sesuai dengan spesifikasi.
4. Mahasiswa mampu memahami penggunaan *return value* dalam fungsi untuk mengembalikan hasil perhitungan.
5. Mahasiswa mampu menggunakan fungsi bawaan dalam R untuk berbagai jenis operasi matematika, statistik, dan manipulasi data.
6. Mahasiswa mampu membuat fungsi rekursif untuk menyelesaikan masalah yang melibatkan pengulangan struktur.

Ukuran Dasar Statistik

R sendiri diciptakan oleh Ross Ihaka dan Robert Gentleman pada tahun 1995. R programming adalah sistem perangkat lunak yang dirancang secara khusus untuk mengerjakan semua hal yang berkaitan dengan statistik. Saat ini R ini menjadi pilihan bagi para peneliti maupun praktisi di berbagai untuk mengolah dan menganalisis data baik untuk kepentingan penelitian maupun bisnis. Untuk mempelajari R programming, terlebih dahulu mengetahui fitur-fitur dasar dalam R.

Ukuran dasar statistik dalam R merujuk pada berbagai metode dan fungsi yang digunakan untuk melakukan perhitungan statistik deskriptif, yaitu ukuran-ukuran yang memberikan gambaran awal mengenai dataset. Berikut adalah beberapa ukuran dasar yang umum digunakan:

1. **Mean (Rata-rata):** Ukuran pusat yang dihitung dengan menjumlahkan semua nilai dan membaginya dengan jumlah observasi. Di R, ini dapat dihitung menggunakan fungsi `mean()`.
 - Contoh: `mean(data$nilai)`
2. **Median (Nilai Tengah):** Nilai tengah dari data yang telah diurutkan. Ini sering digunakan jika data mengandung outlier yang mempengaruhi mean. Dihitung dengan fungsi `median()`.
 - Contoh: `median(data$nilai)`
3. **Modus:** Nilai yang paling sering muncul dalam dataset. R tidak memiliki fungsi bawaan langsung untuk modus, tetapi dapat dihitung dengan membuat fungsi khusus atau menggunakan paket tambahan.
4. **Range (Rentang):** Perbedaan antara nilai maksimum dan minimum dalam dataset. Ini memberikan gambaran kasar tentang sebaran data. Dapat dihitung menggunakan `range()` untuk mendapatkan nilai minimum dan maksimum, atau dengan `diff(range())` untuk mendapatkan rentangnya.
 - Contoh: `range(data$nilai)`
5. **Variance (Varians):** Mengukur seberapa jauh nilai-nilai dalam dataset tersebar dari rata-rata. Varians dapat dihitung dengan fungsi `var()`.
 - Contoh: `var(data$nilai)`
6. **Standard Deviation (Standar Deviasi):** Akar kuadrat dari varians, digunakan untuk memahami penyebaran nilai dalam dataset. Ini dapat dihitung dengan `sd()`.
 - Contoh: `sd(data$nilai)`
7. **Quartiles (Kuartil):** Pembagian data menjadi empat bagian yang sama besar, sering digunakan untuk menggambarkan distribusi data. Kuartil pertama (Q1) mewakili 25% data

pertama, median (Q2) mewakili 50%, dan kuartil ketiga (Q3) mewakili 75%. Fungsi `quantile()` digunakan untuk menghitung kuartil.

- o Contoh: `quantile(data$nilai)`

8. **Interquartile Range (IQR):** Perbedaan antara kuartil ketiga (Q3) dan kuartil pertama (Q1), memberikan gambaran tentang sebaran data tanpa memperhatikan outlier. Ini dihitung dengan `IQR()`.

- o Contoh: `IQR(data$nilai)`

9. **Minimum dan Maksimum:** Nilai terkecil dan terbesar dalam dataset, dihitung dengan `min()` dan `max()`.

- o Contoh: `min(data$nilai), max(data$nilai)`

```
nilai.mean<-mean(x)
nilai.mean
```

```
## [1] 12.24615
```

```
nilai.range<-range(x)
nilai.range
```

```
## [1] -21 54
```

```
nilai.median<-median(x)
nilai.median
```

```
## [1] 8
```

```
nilai.var<-var(x)
nilai.var
```

```
## [1] 409.7544
```

Operator Perbandingan

Operator Perbandingan dalam R digunakan untuk membandingkan dua nilai atau lebih, dan hasil dari perbandingan ini adalah nilai logika berupa TRUE atau FALSE. Operator ini sangat berguna

dalam pengambilan keputusan, seperti kondisi dalam perulangan dan seleksi (*if-else*). Berikut adalah beberapa operator perbandingan dalam R yaitu:

Operator	Logika
"&"	dan (AND)
" "	atau (OR)
"!"	negasi (NOT)
"=="	sama dengan (EQUAL)
"!="	tidak sama dengan (NOT EQUAL)
"%in%"	ada diantara (IN)
">"	lebih besar dari (GREATER)
"<"	lebih kecil dari (LESS)
">="	lebih besar dari atau sama dengan (GREATER OR EQUAL)
"<="	lebih kecil dari atau sama dengan (LESS OR EQUAL)
"&&"	dan (AND) (vector)
" "	atau (OR) (vector)

Contoh:

```
a<-10  
b<-9  
a>b #Lebih besar
```

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

```
a<b #Lebih kecil
```

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

```
a==b #sama dengan
```

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

```
a!=b #tidak sama dengan
```

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

```
a>b #lebih besar sama dengan
```

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

```
b<a #lebih kecil sama dengan
```

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

```
p <- c(TRUE,TRUE, FALSE)  
q <- c(FALSE,TRUE,TRUE)
```

```
print(p&&q) # AND
```

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

```
print(p|q) # OR
```

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

```
print(!p) # NOT
```

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

```
print(p&q) # AND element
```

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

```
print(p|q) # OR element
```

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

Tipe Data

```
jeruk<-TRUE # logical  
class(jeruk)
```

```
## [1] "logical"
```

```
n<-10.9 # numeric  
class(n)
```

```
## [1] "numeric"
```

```
u <- 745i #complex
class(u)
```

```
## [1] "complex"
```

```
k <- "kamu" #character
class(k)
```

```
## [1] "character"
```

```
bil <- "165" #character
class(bil)
```

```
## [1] "character"
```

```
is.character (bil)
```

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

```
is.logical(k)
```

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

class() digunakan untuk melihat termasuk dalam tipe data apa. Selain menggunakan fungsi class(), dapat pula menggunakan fungsi is.numeric(), is.character(), is.logical(), dan sebagainya berdasarkan jenis data apa yang ingin dicek.

Data Frame

```
nama <- c("Abdullah", "Wahsyi", "Sumayyah", "Shafiyah", "Ieran")
nama
```

```
## [1] "Abdullah" "Wahsyi" "Sumayyah" "Shafiyah" "Ieran"
```

```
usia <- c(33, 9, 15, 19, 17)
usia
```

```
## [1] 33 9 15 19 17
```

```
tinggibadan <- c(176, 100, 150, 151, 160)
tinggibadan
```

```
## [1] 176 100 150 151 160
```

```
dataf <- data.frame(nama, usia, tinggibadan)
dataf
```

```
##      nama  usia  tinggibadan
## 1 Abdullah  33      176
## 2 Wahsyi    9      100
## 3 Sumayyah  15      150
## 4 Shafiyah  19      151
## 5 Ieran     17      160
```

Input Data ke R

```
library(readxl)
dataq = read_excel("E:/housing-prices.xlsx")
View(dataq) # melihat data
head(dataq) # menampilkan data pertama
```

```
## # A tibble: 6 x 16
##   Price lot.Size Waterfront Age Land.Value New.Construct Central.Air
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 132500  0.09      0    42  50000      0      0
## 2 181115  0.02      0     0  21500      0      0
## 3 100000  0.19      0   133    7000      0      0
## 4 155000  0.41      0    15  18700      0      0
## 5  80000  0.11      0     0  15000      1      1
## 6 120000  0.68      0    31  14000      0      0
## # ... with 0 more variables: Fuel.Type <chr>, Heat.Type <chr>,
## #   Sewer.Type <chr>, Living.Area <dbl>, Pct.College <dbl>, Bedrooms <dbl>,
## #   Fireplaces <dbl>, Bathrooms <dbl>, Rooms <dbl>
```

```
attach(dataq)
str(dataq) # struktur data
```

```
## tibble [1,734 x 16] (53: tbl_df/tbl/data.frame)
## $ Price      : num [1:1734] 132500 181115 100000 155000 80000 ...
## $ lot.Size   : num [1:1734] 0.09 0.02 0.19 0.41 0.11 0.08 0.4 1.31 0.83 1.94 ...
## $ Waterfront : num [1:1734] 0 0 0 0 0 0 0 0 0 ...
## $ Age        : num [1:1734] 42 0 133 13 0 31 33 23 36 4 ...
## $ Land.Value : num [1:1734] 50000 22000 7000 18700 15000 14000 21500 14000 22200 11200 ...
## $ New.Construct: num [1:1734] 0 0 0 0 1 0 0 0 0 ...
## $ Central.Air : num [1:1734] 0 0 0 0 1 0 0 0 0 ...
## $ Fuel.Type   : chr [1:1734] "Electric" "Gas" "Gas" "Gas" ...
## $ Heat.Type   : chr [1:1734] "Electric" "Hot water" "Hot water" "Hot Air" ...
## $ Sewer.Type  : chr [1:1734] "Private" "Private" "Public" "Private" ...
## $ Living.Area : num [1:1734] 906 1953 1944 1944 840 ...
## $ Pct.College : num [1:1734] 35 51 51 51 51 22 51 35 51 40 ...
## $ Bedrooms   : num [1:1734] 3 3 4 3 2 4 4 4 3 ...
## $ Fireplaces  : num [1:1734] 1 0 1 1 0 1 1 1 0 ...
## $ Bathrooms  : num [1:1734] 1 2.5 1 1.5 1 1 1.5 1.5 1.5 ...
## $ Rooms      : num [1:1734] 5 6 6 5 3 8 8 9 6 ...
```

```
summary(dataq)
```

```
##   Price      lot.Size      Waterfront      Age
## Min.   : 5000   Min.   : 0.0000   Min.   :0.000000   Min.   : 0.00
## 1st Qu.:145000   1st Qu.: 0.1700   1st Qu.:0.000000   1st Qu.: 15.00
## Median :189700   Median : 0.3700   Median :0.000000   Median : 18.00
## Mean   :211545   Mean    : 0.5003   Mean    :0.000251   Mean    : 28.26
## 3rd Qu.:257290   3rd Qu.: 0.5400   3rd Qu.:0.000000   3rd Qu.: 34.00
## Max.   :775000   Max.    :12.2000   Max.    :1.000000   Max.    :125.00
```

```
var(dataq$Price) # nilai variansi pada variabel Price
```

```
## [1] 9712893383
```

```
# pada summary() akan menampilkan ringkasan statistika pada setiap variabel meliputi min, max, mean, median, q1 dan q3.
```

LATIHAN

1. Membandingkan bilangan bulat

```
1 # Membandingkan dua bilangan bulat
2 bilangan1 <- as.integer(readline(prompt = "Input bilangan pertama: "))
3 bilangan2 <- as.integer(readline(prompt = "Input bilangan kedua: "))
4
5 cat("Apakah kedua bilangan sama?", bilangan1 == bilangan2, "\n")
6 cat("Apakah bilangan pertama lebih besar dari bilangan kedua?",
7     bilangan1 > bilangan2, "\n")
8 cat("Apakah bilangan pertama lebih kecil dari bilangan kedua?",
9     bilangan1 < bilangan2, "\n")
```

2. Menghitung Usia dengan Operator Perbandingan

```
1 # Memeriksa apakah usia sudah cukup untuk memilih
2 usia <- as.integer(readline(prompt = "Masukkan usia Anda: "))
3
4 if (usia >= 17) {
5   cat("Anda sudah cukup umur untuk memilih.\n")
6 } else {
7   cat("Anda belum cukup umur untuk memilih.\n")
8 }
```

3. Memeriksa Bilangan Genap atau Ganjil

```
1 # Memeriksa apakah bilangan genap atau ganjil
2 bilangan <- as.integer(readline(prompt = "Input bilangan: "))
3
4 if (bilangan %% 2 == 0) {
5   cat("Bilangan tersebut adalah genap.\n")
6 } else {
7   cat("Bilangan tersebut adalah ganjil.\n")
8 }
```

4. Memeriksa Suhu dalam Rentang Tertentu

```
1 # Memeriksa kategori suhu
2 suhu <- as.numeric(readline(prompt = "Masukkan suhu dalam derajat
3     Celsius: "))
4
5 if (suhu < 0) {
6   cat("Suhu sangat dingin.\n")
7 } else if (suhu >= 0 && suhu <= 25) {
8   cat("Suhu dingin.\n")
9 } else {
10  cat("Suhu panas.\n")
11 }
```

5. Membandingkan Nilai dalam Data Frame

```
1 # Membuat data frame dan membandingkan kolom nilai
2 data <- data.frame(nilai1 = c(10, 5, 8),
3                   nilai2 = c(7, 12, 8))
4
5 hasil <- data$nilai1 > data$nilai2
6 print(hasil)
```

6. Menentukan Kelulusan Berdasarkan Nilai

```
1 # Menentukan kelulusan
2 nilai <- as.numeric(readline(prompt = "Masukkan nilai siswa: "))
3
4 if (nilai >= 60) {
5   cat("Lulus\n")
6 } else {
7   cat("Gagal\n")
8 }
```

7. Menghitung Diskon Barang

```
1 # Menghitung diskon
2 total_belanja <- as.numeric(readline(prompt = "Masukkan total belanja:
3   "))
4
5 if (total_belanja > 100000) {
6   total_bayar <- total_belanja * 0.9
7   cat("Anda mendapatkan diskon. Jumlah yang harus dibayar:",
8     total_bayar, "\n")
9 } else {
10  cat("Anda tidak mendapatkan diskon. Jumlah yang harus dibayar:",
11    total_belanja, "\n")
12 }
```