UNIVERSITAS SULTAN AGENG TIRTAYASA

UNTIRTA Jawara
Jujur Adil Wibawa Amanah Religius Akuntabel

Pengantar Komputasi Statistika

#12 Meeting

Text Manipulation

Ferdian Bangkit Wijaya, S.Stat., M.Si
NIP. 19900520202406100 1

String

String (atau character) adalah tipe data yang digunakan untuk menyimpan teks.

- Dibuat dengan membungkus teks dalam tanda kutip ganda ("...") atau tunggal ('...').
- Spasi, angka, dan simbol di dalam tanda kutip semuanya dianggap sebagai bagian dari teks.

Fungsi nchar() (Number of Characters) Menghitung jumlah karakter dalam sebuah string (termasuk spasi!).

```
teks_1 <- "Statistika"
teks_2 <- "Hello World!"
teks_3 <- "12345"

nchar(teks_1)
# [1] 10

nchar(teks_2)
# [1] 12  (Spasi dan tanda seru dihitung)

nchar(teks_3)
# [1] 5   (Ini adalah TEKS "12345", bukan angka 12345)
```

Whitespace

Spasi yang tidak diinginkan di awal atau akhir string adalah masalah umum saat impor data.
Fungsi trimws() (Trim Whitespace) Menghapus spasi (dan tab, newline) dari kedua sisi (awal dan akhir) string.

```
# Data kotor dengan spasi tidak konsisten
data_kotor <- "  Budi Hartono   "

nchar(data_kotor)
# [1] 17

# Membersihkan spasi
data_bersih <- trimws(data_kotor)
print(data_bersih)
# [1] "Budi Hartono"

nchar(data_bersih)
# [1] 12
# (Spasi di tengah tidak dihapus, hanya di awal/akhir)
```

Vector & Matrix String

Sama seperti numeric, string juga dapat disimpan dalam struktur data yang lebih besar.

Vektor String Dibuat dengan fungsi c().
Matriks String Dibuat dengan fungsi matrix().

```
vektor_nama <- c("Budi", "Ani", "Citra", "Doni")
print(vektor_nama)
# [1] "Budi"  "Ani"   "Citra" "Doni"

# Mengambil elemen ke-2
vektor_nama[2]
# [1] "Ani"
```

```
matriks_kata <- matrix(
  c("A", "B", "C", "D", "E", "F"),
  nrow = 2,
  ncol = 3
)
print(matriks_kata)
#      [,1] [,2] [,3]
# [1,] "A"  "C"  "E"
# [2,] "B"  "D"  "F"
```

Konversi ke String

Kadang kita perlu mengubah angka atau logical menjadi teks.
Fungsi as.character() Memaksa (coerce) tipe data lain menjadi character.

```
angka <- 1945
logika <- TRUE
vektor_angka <- c(10, 20, 30)

# Konversi angka
teks_angka <- as.character(angka)
print(teks_angka)
# [1] "1945"
```

```
# Konversi logical
teks_logika <- as.character(logika)
print(teks_logika)
# [1] "TRUE"

# Konversi seluruh vektor
teks_vektor <- as.character(vektor_angka)
print(teks_vektor)
# [1] "10" "20" "30"
```

Konversi String ke Tipe LAIN

Ini adalah operasi yang sangat umum setelah mengimpor data.
Fungsi as.numeric() dan as.logical() Mengubah string yang terlihat seperti angka/logika.
PENTING: Apa yang Terjadi Jika Gagal? Jika R tidak bisa mengonversi string (misal: "Budi" jadi angka), R akan menghasilkan NA (Not Available / Missing).

```
# String angka
str_angka <- "123.5"
str_angka_salah <- "100ribu"
str_logika <- "FALSE"

# Konversi ke Numeric (Berhasil)
angka_nyata <- as.numeric(str_angka)
angka_nyata + 10
# [1] 133.5
```

```
# Konversi ke Numeric (Gagal -> NA)
angka_gagal <- as.numeric(str_angka_salah)
print(angka_gagal)
# [1] NA
# Warning message: NAs introduced by coercion

# Konversi ke Logical (Berhasil)
logika_nyata <- as.logical(str_logika)
print(logika_nyata)
# [1] FALSE
```

Latihan 1

Tugas:

1. Buat vektor vektor_nilai_kotor <- c(" 90", "75 ", " 82.5 ", "Tugas_Selesai").
2. Bersihkan spasi di awal dan akhir dari vektor tersebut.
3. Konversi vektor yang sudah bersih menjadi numeric.
4. Hitung mean() (rata-rata) dari data numerik tersebut. (Ingat na.rm!)

Jawaban Latihan 1

```
# 1. Vektor kotor
vektor_nilai_kotor <- c(" 90", "75 ", " 82.5 ", "Tugas_Selesai")

# 2. Bersihkan spasi
vektor_bersih <- trimws(vektor_nilai_kotor)
print(vektor_bersih)
# [1] "90"            "75"            "82.5"          "Tugas_Selesai"

# 3. Konversi ke numeric
vektor_numerik <- as.numeric(vektor_bersih)
print(vektor_numerik)
# [1] 90.0 75.0 82.5   NA
# (Tugas_Selesai menjadi NA)

# 4. Hitung rata-rata, abaikan NA
mean(vektor_numerik, na.rm = TRUE)
# [1] 82.5
```

Penggabungan String

paste() (Menggabungkan dengan spasi/pemisah kustom)

- sep = " " (separator/pemisah) adalah default-nya (spasi).

paste0() (Menggabungkan tanpa spasi)

- Versi cepat dari paste(..., sep = "").

```
kata_1 <- "Fakultas"
kata_2 <- "Teknik"

# Default 'paste' (pemisah spasi)
paste(kata_1, kata_2)
# [1] "Fakultas Teknik"

# 'paste' dengan pemisah kustom
paste(kata_1, kata_2, sep = "-")
# [1] "Fakultas-Teknik"
```

```
# 'paste0' (tanpa pemisah)
paste0(kata_1, kata_2)
# [1] "FakultasTeknik"

# Menggabungkan elemen-elemen vektor (collapse)
vektor_kata <- c("satu", "dua", "tiga")
paste(vektor_kata, collapse = ", ")
# [1] "satu, dua, tiga"
```

Pemisahan String

Memecah satu string menjadi beberapa bagian berdasarkan delimiter (karakter pemisah).
PENTING: strsplit() selalu mengembalikan LIST! Karena ia dirancang untuk bisa memecah vektor string sekaligus.

```
data_csv <- "Budi,85,Jakarta"
data_npm <- "22-01-30-1005"

# Memecah berdasarkan koma (,)
hasil_split_1 <- strsplit(data_csv, split = ",")
print(hasil_split_1)
# [[1]]  <- Ini adalah LIST
# [1] "Budi"    "85"      "Jakarta"

# Mengambil elemen vektornya
hasil_vektor_1 <- hasil_split_1[[1]]
print(hasil_vektor_1[2])
# [1] "85"
```

```
# Memecah berdasarkan strip (-)
hasil_vektor_2 <- strsplit(data_npm, split = "-")[[1]]
print(hasil_vektor_2)
# [1] "22" "01" "30" "1005"
```

Substring

Mengekstrak (memotong) bagian dari string berdasarkan posisi karakter.
Struktur: substr(teks, start = posisi_mulai, stop = posisi_akhir)

```
teks <- "KomputasiStatistika"

# Mengambil 4 karakter pertama
substr(teks, start = 1, stop = 4)
# [1] "Komp"

# Mengambil kata "Statistika" (posisi 10 s/d 19)
substr(teks, start = 10, stop = 19)
# [1] "Statistika"

# Mengambil 3 karakter terakhir
# (nchar() bisa dipakai untuk menghitung akhir)
substr(teks, start = nchar(teks) - 2, stop = nchar(teks))
# [1] "ika"
```

Latihan 2

Tugas:

1. Buat variabel id_mhs <- "2024-10015020".
2. Gunakan strsplit() untuk memisahkan Tahun dan NPM (berdasarkan -). Simpan hasilnya di list_hasil.
3. Ekstrak Tahun (elemen 1) dan NPM (elemen 2) dari list_hasil ke variabel baru.
4. Gunakan substr() pada NPM (dari langkah 3) untuk mengambil 4 digit terakhir saja (kode unik).
5. Gunakan paste0() untuk membuat "Username" baru dengan format: TahunKodeUnik (Contoh: "20245020").

Jawaban Latihan 2

```
# 1. Data awal
id_mhs <- "2024-10015020"

# 2. Split (menghasilkan list)
list_hasil <- strsplit(id_mhs, split = "-")

# 3. Ekstrak
tahun <- list_hasil[[1]][1]  # [1] "2024"
npm <- list_hasil[[1]][2]    # [1] "10015020"

# 4. Ambil 4 digit terakhir NPM
kode_unik <- substr(npm, start = nchar(npm) - 3, stop = nchar(npm))
# [1] "5020"

# 5. Gabungkan
username <- paste0(tahun, kode_unik)
print(username)
# [1] "20245020"
```

Membaca File Teks

Cara paling dasar untuk membaca file teks (.txt, .log, .csv mentah) di R.
Fungsi readLines()

- Membaca file baris demi baris.
- Setiap baris di file menjadi satu elemen dalam vektor character.

File 1: puisi.txt
Di atas langit
Ada langit
Selalu rendah hati

File 2: nilai_mentah.txt
80
75
 90
65
55

```
# Membaca file 'puisi.txt'
data_puisi <- readLines("puisi.txt")

# Cek hasilnya (Vektor character)
print(data_puisi)
# [1] "Di atas langit"
# [2] "Ada langit"
# [3] "Selalu rendah hati"

# Cek jumlah baris
length(data_puisi)
# [1] 3
```

Meringkas Teks

Kita bisa menggabungkan semua fungsi yang telah dipelajari untuk melakukan analisis teks sederhana, seperti menghitung frekuensi kata.

```
# 1. Gabungkan semua baris jadi 1 teks panjang
teks_lengkap <- paste(data_puisi, collapse = " ")
# [1] "Di atas langit Ada langit Selalu rendah hati"

# 2. Ubah ke huruf kecil (agar 'Langit' dan 'langit' sama)
teks_lengkap <- tolower(teks_lengkap)
# [1] "di atas langit ada langit selalu rendah hati"

# 3. Pisahkan setiap kata (pemisah spasi)
list_kata <- strsplit(teks_lengkap, split = " ")

# 4. Ubah list menjadi vektor
vektor_kata <- list_kata[[1]]
# [1] "di" "atas" "langit" "ada" "langit" "selalu" "rendah" "hati"

# 5. Hitung frekuensi kata menggunakan 'table()'
table(vektor_kata)
```

Output :

```
vektor_kata
ada atas di hati langit rendah selalu
1 1 1 1 2 1 1
```

Latihan 3

Tugas:

1. Gunakan readLines() untuk membaca nilai_mentah.txt ke dalam variabel data_txt.
2. Periksa data_txt. Anda akan lihat beberapa elemen punya spasi (" 90").
3. Gunakan trimws() untuk membersihkan spasi dari data_txt.
4. Gunakan as.numeric() untuk mengubah data bersih menjadi vektor angka.
5. Hitung mean() dan max() dari nilai-nilai tersebut.

Jawaban Latihan 3

```
# 1. Baca file
data_txt <- readLines("nilai_mentah.txt")
print(data_txt)
# [1] "80"   "75"   " 90"  "65 "  "55"

# 3. Bersihkan spasi (Langkah 2 adalah inspeksi)
data_bersih <- trimws(data_txt)
print(data_bersih)
# [1] "80" "75" "90" "65" "55"

# 4. Konversi ke numeric
data_numerik <- as.numeric(data_bersih)
print(data_numerik)
# [1] 80 75 90 65 55

# 5. Hitung statistik
mean(data_numerik)
# [1] 73

max(data_numerik)
# [1] 90
```

UNIVERSITAS SULTAN AGENG TIRTAYASA

UNTIRTA Jawara
Jujur Adil Wibawa Amanah Religius Akuntabel

SEE YOU NEXT WEEK !

Ferdian Bangkit Wijaya, S.Stat., M.Si
NIP. 19900520202406100 1
ferdian.bangkit@untirta.ac.id