

# COMPUTER PROGRAMING 2

FORMAT STRING

MODUL 4

2024

## DAFTAR ISI

|                                                                   |   |
|-------------------------------------------------------------------|---|
| DAFTAR ISI .....                                                  | 2 |
| FORMAT STRING .....                                               | 3 |
| A. Format String .....                                            | 3 |
| 1. Memformat String dengan Format Specifiers.....                 | 3 |
| 2. Memformat String dengan Fungsi format() .....                  | 3 |
| 3. Memformat String dengan String Interpolation / F-Strings ..... | 4 |
| B. Memecah String.....                                            | 4 |
| C. Mengubah Case String .....                                     | 5 |
| 1. Mengubah String Menjadi Upper Case .....                       | 5 |
| 2. Mengubah String Menjadi Lower Case .....                       | 6 |
| 3. Mengubah String Menjadi Reverse Case .....                     | 6 |
| D. Mencari Kata Pada String .....                                 | 7 |
| E. Mengganti Kata pada String.....                                | 7 |
| F. Menghapus Karakter Tertentu Pada String.....                   | 7 |
| G. Menghitung Jumlah Karakter Yang Muncul .....                   | 8 |

## FORMAT STRING

---

### A. Format String

String formatting pada python adalah satu proses memasukkan atau menyisipkan variabel atau nilai ke dalam template string yang telah ditentukan. Terdapat setidaknya 3 cara untuk melakukan hal ini. Beberapa di antaranya adalah cara jadul, dan beberapa di antaranya adalah cara yang lebih baru dan modern.

#### 1. Memformat String dengan Format Specifiers

Cara yang pertama (dan ini adalah cara yang cukup lama), adalah menggunakan format specifiers. Format specifiers adalah simbol **%s** yang berarti string.

Contoh:

```
kata = 'Halo, saya %s dari %s'

print(kata % ('Lendis Fabri', 'Indonesia'))
```

Output:

```
Halo, saya Lendis Fabri dari Indonesia
```

#### 2. Memformat String dengan Fungsi format()

Untuk cara yang kedua adalah menggunakan fungsi `string.format()` yang baru diperkenalkan pada python versi 3

Berikut ini contoh penggunaannya:

```
kata1 = 'Halo, saya {nama} dari {asal}'
kata2 = 'Saya suka makan {} dan minum {}'
print(kata1.format(nama = 'Lendis Fabri', asal = 'Indonesia'))
print(kata2.format('Nasi Goreng', 'air putih'))
```

Outputnya:

```
Halo, saya Lendis Fabri dari Indonesia

Saya suka makan Nasi Goreng dan minum air putih
```

### 3. Memformat String dengan String Interpolation / F-Strings

String interpolation atau F-Strings adalah fitur terbaru dari python 3.6. Fungsi ini memiliki tujuan yang sama dengan 2 cara sebelumnya. Hanya saja, ia memiliki sintaks yang jauh lebih modern dan lebih sederhana.

Hal ini akan terlihat sangat familiar bagi kalian yang pernah belajar javascript mau pun kotlin, karena secara sintaks penulisannya hampir sama dengan string template literal pada dua bahasa tersebut.

Berikut ini contohnya:

```
nama = 'Lendis Fabri'
asal = 'Indonesia'

print(f'Perkenalkan saya {nama} dari {asal}')
```

Output:

```
Perkenalkan saya Lendis Fabri dari Indonesia
```

Bagaimana? Jauh lebih manusiawi bukan sintaksnya?

## B. Memecah String

Fungsi selanjutnya adalah memecah string menjadi list. Hal ini bisa kita lakukan menggunakan fungsi **string.split()** dan **string.join()**.

Contoh:

```
alamat = 'Surabaya, Jawa Timur, Indonesia'

print(alamat.split())
print(alamat.split(','))
print(alamat.split(', '))
print('❤️'.join(['aku', 'suka', 'python']))
print('👉'.join(alamat.split(', ')))
```

Output:

```
['Surabaya,', 'Jawa', 'Timur,', 'Indonesia']  
['Surabaya', ' Jawa Timur', ' Indonesia']  
['Surabaya', 'Jawa Timur', 'Indonesia']  
aku♡suka♡python  
Surabaya🦁Jawa Timur🦁Indonesia
```

Penjelasan

- Fungsi `string.split()` menerima satu parameter optional. Parameter ini akan dijadikan sebagai pembatas atau separator yang kemudian string akan dibagi menjadi list berdasarkan pembatas tersebut.
- Jika fungsi `string.split()` tidak diberi parameter, maka defaultnya adalah spasi ( ).
- Fungsi `string.join()` berfungsi untuk menggabungkan item list menjadi string utuh, dengan string awal sebagai glue atau perekat antar masing-masing item

### C. Mengubah Case String

Contoh kasus yang akan kita lakukan pada materi mengubah case pada string terdiri dari 3 contoh diantaranya yaitu: mengubah string menjadi upper case, mengubah string menjadi lower case dan mengubah string menjadi reverse case

#### 1. Mengubah String Menjadi Upper Case

Selanjutnya kita bisa memodifikasi string menjadi uppercase alias huruf besar semua dengan fungsi `upper()`.

Contoh:

```
print('halo selamat pagi!'.upper())  
print('Halo Selamat Siang!'.upper())
```

Output:

```
HALO SELAMAT PAGI!  
HALO SELAMAT SIANG!
```

## 2. Mengubah String Menjadi Lower Case

Kebalikannya, kita bisa menggunakan fungsi `string.lower()` untuk mengubah string menjadi huruf kecil semua.

Contoh:

```
print('APA KABAR?'.lower())  
print('Lagi Dimana?'.lower())
```

Output:

```
apa kabar?  
lagi dimana?
```

Selanjutnya, untuk mengubah `teks string` menjadi `Teks String` alias **title case**, kita bisa menggunakan fungsi `string.title()`.

Perhatikan contoh berikut:

```
print('daun tidak pernah membenci angin'.title())  
print('LASKAR PELANGI'.title())
```

Output;

```
Daun Tidak Pernah Membenci Angin  
Laskar Pelangi
```

## 3. Mengubah String Menjadi Reverse Case

Yang terakhir, kita bisa membalikkan huruf kecil menjadi huruf besar, dan huruf besar menjadi huruf kecil dengan perintah `string.swapcase()`.

Perhatikan contoh:

```
print('Indonesia Adalah Negara Kepulauan'.swapcase())  
print('aKu cInTa NeGeRiKu'.swapcase())
```

Output:

```
iNDONESIA aDALAH nEGARA KEPULAUAN  
AkU CiNtA nEgErIKU
```

#### D. Mencari Kata Pada String

Pada python, mencari sebuah kata atau karakter dalam suatu string caranya sangat mudah. Kita bisa menggunakan fungsi `string.find()`.

Fungsi tersebut akan mengembalikan indeks dari hasil pertama pencarian, dan akan mengembalikan -1 jika karakter yang dicari tidak ditemukan.

Contoh:

```
angka = '12345'

print('Index angka 3:', angka.find('3'))
print('Index angka 45:', angka.find('45'))
print('Index angka 35:', angka.find('35'))
```

Output:

```
Index angka 3: 2
Index angka 45: 3
Index angka 35: -1
```

#### E. Mengganti Kata pada String

Dengan memanfaatkan fungsi `string.replace()`, kita bisa mencari lalu menimpa sebuah karakter/teks dengan teks baru.

Perhatikan contoh berikut:

```
ibu_kota = 'Jakarta'

print(ibu_kota.replace('a', 'i'))
print(ibu_kota.replace('kar', '🐍'))
```

Output:

```
Jikirti
Ja🐍ta
```

#### F. Menghapus Karakter Tertentu Pada String

Tentu saja, jika kita memanggil fungsi `replace()` pada suatu string, lalu kita timpa teks lama dengan string kosong ("), seolah-olah kita telah menghapus karakter tersebut.

Contoh:

```
ibu_kota = 'Jakarta'
print(ibu_kota.replace('a', ''))
```

Output:

```
Jkrt
```

## G. Menghitung Jumlah Karakter Yang Muncul

Terakhir tapi bukan yang paling akhir, kita bisa menghitung berapa kali sebuah teks muncul dalam suatu string.

Misal kita ingin menghitung berapa kali kata di muncul pada sebuah paragraf, kita bisa menggunakan fungsi `count()` pada string.

Perhatikan contoh berikut:

```
paragraf = 'Baru-baru ini telah ditemukan sebuah meteor di
suatu sungai'
x = paragraf.count('di')

print(f'Kata "di" muncul sebanyak {x} kali')
```

Output:

```
Kata "di" muncul sebanyak 2 kali
```

=== To be Continue ===