

# Input atau Output String

Oleh: NURAZIZAH

# String dalam pemrograman

String adalah salah satu tipe data yang paling sering digunakan. String memungkinkan kita untuk merepresentasikan teks, seperti kata, kalimat, atau bahkan paragraf.

# Apa itu String?

String adalah urutan karakter yang diapit oleh tanda kutip. Dalam kebanyakan bahasa pemrograman, Anda dapat menggunakan tanda kutip tunggal (' ') atau ganda (" ") untuk membuat string.

# Input String

1

Input string adalah proses menerima data teks dari pengguna atau sumber lain.

2

Contoh:

```
nama = input("Masukkan  
nama Anda: ")  
print("Halo, " + nama + "!")
```

# Output String

1

Output string adalah proses menampilkan teks kepada pengguna atau menyimpannya ke sumber lain. Beberapa metode umum untuk menghasilkan output string meliputi:

Contoh:

```
print("Ini adalah output string.")
```

2

# Program Fungsi 7 String

1. Fungsi split (): Memecah string menjadi daftar substring berdasarkan pemisah tertentu (default: spasi).

```
In [1]: def hitung_kata(kalimat):
    kata_kata = kalimat.split()
    return len(kata_kata)

kalimat = "Ini adalah contoh kalimat untuk dihitung jumlah katanya."
jumlah = hitung_kata(kalimat)
print(f"Jumlah kata: {jumlah}")
```

Jumlah kata: 8

## 2

### Fungsi reverse(): Membalikkan urutan elemen dalam daftar.

```
In [2]: def balik_kata(kalimat):
    kata_kata = kalimat.split()
    kata_kata.reverse()
    return " ".join(kata_kata)

kalimat = "Ini adalah contoh kalimat."
kalimat_terbalik = balik_kata(kalimat)
print(f"Kalimat terbalik: {kalimat_terbalik}")
```

Kalimat terbalik: kalimat. contoh adalah Ini

## Bab 3

3

Fungsi join(): Menggabungkan elemen-elemen daftar menjadi string dengan pemisah tertentu.

```
In [7]: def ganti_kata(kalimat, kata_lama, kata_baru):
    return kalimat.replace(kata_lama, kata_baru)

kalimat = "Saya suka apel."
kalimat_baru = ganti_kata(kalimat, "apel", "pisang")
print(f"Kalimat baru: {kalimat_baru}")
```

Kalimat baru: Saya suka pisang.

## Bab 3

### 4

## Fungsi replace(): Mengganti substring dengan substring lain.

```
In [6]: def cek_palindrom(teks):
    teks = teks.lower().replace(" ", "")
    return teks == teks[::-1]

teks1 = "kasur rusak"
teks2 = "bukan palindrom"
print(f'{teks1} adalah palindrom: {cek_palindrom(teks1)}')
print(f'{teks2} adalah palindrom: {cek_palindrom(teks2)})
```

```
'kasur rusak' adalah palindrom: True
'bukan palindrom' adalah palindrom: False
```

## Bab 3

5

Fungsi lower(): Mengubah string menjadi huruf kecil.

```
In [5]: def hitung_karakter(teks, karakter):
    return teks.lower().count(karakter.lower())

teks = "Ini adalah contoh teks."
karakter = "a"
jumlah = hitung_karakter(teks, karakter)
print(f"Jumlah karakter '{karakter}': {jumlah}")
```

Jumlah karakter 'a': 3

# Bab 3

## 6

### Fungsi count(): Menghitung jumlah kemunculan substring.

```
In [4]: def ubah_format_tanggal(tanggal_lama):
    from datetime import datetime
    tanggal_obj = datetime.strptime(tanggal_lama, "%Y-%m-%d")
    return tanggal_obj.strftime("%d/%m/%Y")

tanggal_lama = "2023-10-27"
tanggal_baru = ubah_format_tanggal(tanggal_lama)
print(f"Tanggal baru: {tanggal_baru}")
```

Tanggal baru: 27/10/2023

## Bab 3

7

Fungsi `re.findall()`: Mencari semua kemunculan pola dalam string menggunakan regular expression.

```
In [3]: import re

def ekstrak_substring(tekst, pola):
    hasil = re.findall(pola, tekst)
    return hasil

tekst = "Nomor telepon: 0812-3456-7890 dan 0877-1234-5678"
pola = r"\d{4}-\d{4}-\d{4}"
nomor_telepon = ekstrak_substring(tekst, pola)
print(f"Nomor telepon: {nomor_telepon}")
```

```
Nomor telepon: ['0812-3456-7890', '0877-1234-5678']
```

# Pentingnya Input/Output

- Memungkinkan interaksi dengan pengguna.
- Memfasilitasi pemrosesan data teks.
- Memungkinkan penyimpanan dan pengambilan data.



# Terima Kasih