



# Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

<b>NIM</b>	<b>71230983</b>
<b>Nama Lengkap</b>	<b>DELVIN CAPPINOVA</b>
<b>Minggu ke / Materi</b>	<b>12 set</b>

SAYA MENYATAKAN BAHWA LAPORAN PRAKTIKUM INI SAYA BUAT DENGAN USAHA SENDIRI TANPA MENGGUNAKAN BANTUAN ORANG LAIN. SEMUA MATERI YANG SAYA AMBIL DARI SUMBER LAIN SUDAH SAYA CANTUMKAN SUMBERNYA DAN TELAH SAYA TULIS ULANG DENGAN BAHASA SAYA SENDIRI.

SAYA SANGGUP MENERIMA SANKSI JIKA MELAKUKAN KEGIATAN PLAGIASI, TERMASUK SANKSI TIDAK LULUS MATA KULIAH INI.

PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS KRISTEN DUTA WACANA  
YOGYAKARTA  
2024

## BAGIAN 1: MATERI MINGGU INI (40%)

Pada bagian ini, tuliskan kembali semua materi yang telah anda pelajari minggu ini. Sesuaikan penjelasan anda dengan urutan materi yang telah diberikan di saat praktikum. Penjelasan anda harus dilengkapi dengan contoh, gambar/ilustrasi, contoh program (source code) dan outputnya. Idealnya sekitar 5-6 halaman.

### MATERI 1

Set adalah salah satu tipe data pada Python yang dapat digunakan untuk menyimpan sekumpulan data yang semuanya unik. Biasa juga dikenal dengan istilah himpunan

Set memiliki sifat yaitu

- Isi dari Set disebut sebagai anggota (member).
- Anggota dari Set harus bersifat immutable. Beberapa tipe data immutable pada Python: integer, float, string, tuple dan lain-lain. Dengan demikian list dan dictionary (mutable) tidak dapat dimasukkan ke dalam Set.
- Set sendiri bersifat mutable, artinya anda dapat menambah atau mengurangi isi dari sebuah Set. Karena itu Set tidak dapat dimasukkan ke dalam Set.

```
# dengan menggunakan {}
bilangan_genap = {2, 4, 6, 8, 10, 12}
bilangan_ganjil = {1, 3, 5, 7, 9, 11}

# dengan menggunakan fungsi set()
pernah_ke_bulan = set('Neil Armstrong', 'Buzz Aldrin')
```

Set tidak memiliki indeks, karena itu anda tidak dapat mengakses anggota-anggota dari sebuah Set secara langsung menggunakan indeks. Perhatikan contoh program berikut ini:

```
nim = {'71200120', '71200195', '71200214'}
jumlah_nim = len(nim)
print(jumlah_nim)          # akan menghasilkan output 3

# tampilkan isi set satu-persatu
for n in nim:
    print(n)
```

Outputnya adalah

```
3
71200214
71200195
71200120
```

Set adalah tipe data mutable, artinya isinya bisa bertambah atau berkurang. Program berikut ini menunjukkan bagaimana cara menambah anggota sebuah Set dengan fungsi add():

```
# definisikan sebuah set kosong
plat_nomor = set()

# tambahkan plat nomor 'AB 1890 XA'
plat_nomor.add('AB 1890 XA')

# tambahkan plat nomor 'AD 6810 MT'
plat_nomor.add('AD 6810 MT')

# jumlah anggota di dalam Set
print(len(plat_nomor))

# tambahkan plat yang sama sekali lagi
plat_nomor.add('AB 1890 XA')

# tampilkan semua plat nomor
for plat in plat_nomor:
    print(plat)
```

Outputnya adalah

2

AD 6810 MT

AB 1890 XA

Set memiliki mekanisme untuk melakukan pengecekan apakah anggota baru yang akan dimasukkan sudah ada di dalam Set (cek duplikasi). Jika belum ada, maka anggota tersebut bisa masuk ke dalam Set. Tetapi jika ternyata di dalam Set sudah ada anggota dengan nilai yang sama, maka pemanggilan fungsi `add()` tidak akan menambah anggota dari Set. Pengecekan duplikasi ini sudah ada di dalam fungsi `add()`, sehingga anda tidak perlu melakukannya sendiri.

Berikut adalah beberapa fungsi dalam anggota set

<b>discard()</b>	<b>remove()</b>	<b>pop()</b>	<b>clear()</b>
Menghapus satu elemen yang disebutkan	Menghapus satu elemen yang disebutkan	Mengambil salah satu dan menghapusnya dari set (tidak tentu)	Menghapus seluruh elemen di dalam set
Tidak ada error	Muncul error jika elemen yang dihapus tidak ada	Error jika set kosong	Tidak ada error

Berikut adalah contoh penggunaannya

```

print(bilangan_prima)

# hapus 97 (tidak ada)
bilangan_prima.discard(97)
print(bilangan_prima)

# ambil dan hapus salah satu
bilangan = bilangan_prima.pop()
print(bilangan)
print(bilangan_prima)

# kosongkan set
bilangan_prima.clear()
print(bilangan_prima)

```

Dan ini outputnya

```

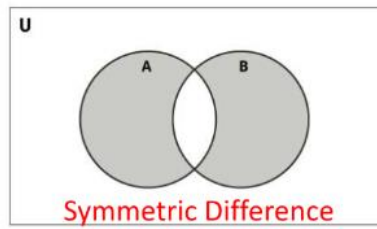
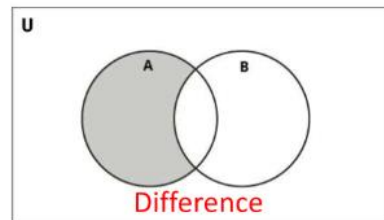
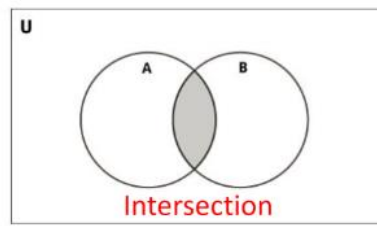
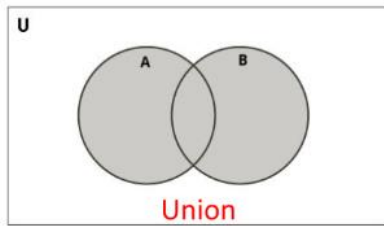
{5, 7, 11, 13, 23, 29}      # awal
{7, 11, 13, 23, 29}        # setelah 5 dihapus. Perhatikan urutan berubah
{7, 11, 13, 23, 29}        # 97 tidak ada, sehingga Set tidak berubah
7                            # fungsi pop() mengeluarkan 7
{11, 13, 23, 29}           # setelah 7 keluar dari Set
set()                       # setelah isi dari Set dihapus semua

```

Operasi-Operasi pada Set Operasi-operasi pada Set adalah operasi-operasi pada himpunan. Berikut ini adalah daftar operasi-operasi Set pada Python:

- Operator Union. Menggabungkan dua Set menjadi satu. Dapat menggunakan operator `|` maupun fungsi `union()`.
- Operator Intersection. Menghasilkan irisan dari dua Set. Dapat menggunakan operator `&` maupun fungsi `intersection()`.
- Operator Difference. Menghasilkan Set baru yang merupakan selisih dari dua Set yang dibandingkan. Dapat menggunakan operator `-` maupun fungsi `difference()`.
- Operator Symmetric Difference. Menghasilkan Set baru yang merupakan jumlah dari dua Set kecuali irisannya. Dapat menggunakan operator maupun fungsi `symmetric_difference()`.

Bentuk contoh operasi pada set dalam bentuk lingkaran



## BAGIAN 2: LATIHAN MANDIRI (60%)

Pada bagian ini anda menuliskan jawaban dari soal-soal Latihan Mandiri yang ada di modul praktikum. Jawaban anda harus disertai dengan source code, penjelasan dan screenshot output.

No 1

```
n = int(input('Masukkan jumlah kategori: '))
data_aplikasi = {}
for i in range(n):
    nama_kategori = input('Masukkan nama kategori:')
    print('Masukkan 5 nama aplikasi di kategori', nama_kategori)
    aplikasi = []
    for j in range(5):
        nama_aplikasi = input('Nama aplikasi: ')
        aplikasi.append(nama_aplikasi)

    data_aplikasi[nama_kategori] = aplikasi

print(data_aplikasi)
daftar_aplikasi_list = []

for aplikasi in data_aplikasi.values():
    daftar_aplikasi_list.append(set(aplikasi))
print(daftar_aplikasi_list)

hasil = daftar_aplikasi_list[0]
for i in range(1, len(daftar_aplikasi_list)):
    hasil = hasil.intersection(daftar_aplikasi_list[i])
print("Aplikasi yang muncul di semua kategori:", hasil)

satu_kategori = set()
for i in range(len(daftar_aplikasi_list)):
    satu_kategori.difference_update(*daftar_aplikasi_list[:i]+daftar_aplikasi_list[i+1:])
print("Aplikasi hanya muncul di satu kategori: ", satu_kategori)
satu_kategori.clear()

dua_kategori = set()
for i in range(len(daftar_aplikasi_list)):
    dua_kategori.intersection_update(daftar_aplikasi_list[i])
print("Aplikasi yang muncul tepat di dua kategori: ", dua_kategori)
dua_kategori.clear()
```

Jadi untuk kode ini adalah update ddari kode dari playstore di materi

Dan hanya menambah output jika ada kategori yang sama

No 2

```
2.py > ...
1 data_list = [1, 2, 3, 4, 5, 5, 5, 6, 6, 6, 7, 8, 8, 9]
2 data_set = {1, 2, 3, 4, 5, 6, 7, 8, 9}
3 data_tuple = (1, 2, 3, 4, 5, 6, 7, 8, 9)
4
5 print("Data list sebelum konversi:", data_list)
6 print("Data set sebelum konversi:", data_set)
7 print("Data tuple sebelum konversi:", data_tuple)
8
9 data_kelist = set(data_list)
10 print("\nData set dari list setelah konversi:", data_kelist)
11
12 data_keset = list(data_set)
13 print("\nData list dari set setelah konversi:", data_keset)
14
15 data_ketuple = set(data_tuple)
16 print("\nData set dari tuple setelah konversi:", data_ketuple)
17
18 data_tuple_set = tuple(data_set)
19 print("\nData tuple dari set setelah konversi:", data_tuple_set)
```

PROBLEMS OUTPUT DEBUG CONSOLE PORTS TERMINAL COMMENTS

Data list dari set setelah konversi: [1, 2, 3, 4, 5, 6, 7, 8, 9]

Data set dari tuple setelah konversi: {1, 2, 3, 4, 5, 6, 7, 8, 9}

Data tuple dari set setelah konversi: (1, 2, 3, 4, 5, 6, 7, 8, 9)

Hanya disuruh untuk melakukan perubahan dari list ke set ke tuple dan tuple ke set

No 3



```

file1 = input("Masukkan nama file pertama: ")
file2 = input("Masukkan nama file kedua: ")

try:
    with open(file1, 'r') as f:
        text1 = f.read().lower()
except FileNotFoundError:
    print(f"File '{file1}' tidak ditemukan.")
    exit()

try:
    with open(file2, 'r') as f:
        text2 = f.read().lower()
except FileNotFoundError:
    print(f"File '{file2}' tidak ditemukan.")
    exit()

kata1 = set(text1.split())
kata2 = set(text2.split())

katakata = kata1.intersection(kata2)

print("Semua kata-kata yang muncul pada kedua file:")
for kata in sorted(katakata):
    print(kata)

```

Ini disuruh untuk membandingkan 2 file txt dan jika salah input akan tertulis salah file