



Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

NIM	71230966
Nama Lengkap	Jevon Christian Putra Funay
Minggu ke / Materi	13 / Set

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%)

Pengenalan Set

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

- 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.

Pengaksesan Set

```
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)
```

Output:

```
3
71200214
71200195
71200120
```

Jika dicermati, output yang dihasilkan urutannya berbeda dengan deklarasi Set sebelumnya. Hal ini disebabkan karena pada Set tidak ada indeks, sehingga tidak ada urutan posisi anggota. Pada tipe data Set, posisi anggota tidaklah penting. 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)

```

Output:

```

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.

Untuk menghapus anggota dari sebuah Set, ada beberapa cara yaitu dengan fungsi `discard()`, `remove()`, `pop()` dan `clear()`. Perbedaan dari empat fungsi tersebut dapat dilihat pada Gambar 12.1.

discard()	remove()	pop()	clear()
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

Gambar 12.1: Fungsi-fungsi untuk menghapus anggota dari Set.

Contoh:

```
nyobba.py > ...
1  bilangan_prima = {13, 23, 7, 29, 11, 5}
2  bilangan_prima.remove(5)
3  print(bilangan_prima)
4  bilangan_prima.discard(97)
5  print(bilangan_prima)
6  bilangan = bilangan_prima.pop()
7  print(bilangan)
8  print(bilangan_prima)
9  bilangan_prima.clear()
10 print(bilangan_prima)
11
```

Output:

```
PS C:\Users\jevon\OneDrive\Documents\Code JAJAJAA
/Users/jevon/OneDrive/Documents/Code JAJAJAAJAJAJ
{29, 23, 7, 11, 13}
{29, 23, 7, 11, 13}
29
{23, 7, 11, 13}
set()
PS C:\Users\jevon\OneDrive\Documents\Code JAJAJAA
```

Contoh lain:

```
nyobba.py > ...
1  ikan = set(['koi', 'koki', 'kembung', 'salmon'])
2  print(ikan)
3  ikan.remove('koi')
4  ikan.add('teri')
5  print(ikan)
```

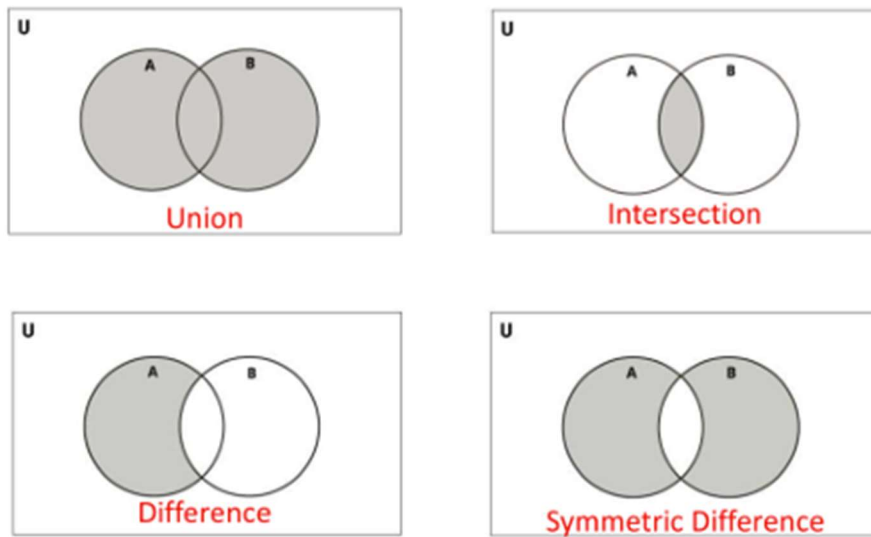
Output:

```
PS C:\Users\jevon\OneDrive\Documents\Code JAJAJAAJAJAJ\PertemuanSET> & C:/Users/jevon/OneDrive/Documents/Code JAJAJAAJAJAJ/PertemuanSET/nyobba.py
{'kembung', 'koki', 'koi', 'salmon'}
{'salmon', 'kembung', 'koki', 'teri'}
PS C:\Users\jevon\OneDrive\Documents\Code JAJAJAAJAJAJ\PertemuanSET>
```

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()



Gambar 12.2: Operasi-operasi pada Set.

Union

```
nyobba.py > ...
1  merek_hp = {'Samsung', 'Apple', 'Xiaomi', 'Sony'}
2  merek_ac = {'LG', 'Samsung', 'Panasonic', 'Daikin', 'Sony'}
3  gabungan = merek_hp | merek_ac
4  print(gabungan)
5
```

```
PS C:\Users\jevon\OneDrive\Documents\Code JAJAJAAJAJAJ\PertemuanSET> & C:/Users/jevon/OneDrive/Documents/Code JAJAJAAJAJAJ/PertemuanSET/nyobba.py
{'Xiaomi', 'LG', 'Sony', 'Panasonic', 'Apple', 'Daikin', 'Samsung'}
PS C:\Users\jevon\OneDrive\Documents\Code JAJAJAAJAJAJ\PertemuanSET>
```

Intersection

```
nyobba.py > ...
1  renang = {'siti', 'mail', 'ikhshan', 'upin', 'ipin'}
2  tenis = {'joko', 'mail', 'ipin', 'upin', 'tejo'}
3  renang_tenis = renang & tenis
4  print(reenang_tenis)
```



```

● PS C:\Users\jevon\OneDrive\Documents\Code JAJAJAAJAJAJ\PertemuanSET> & C:/Users/jevon/OneDrive/Documents/Code JAJAJAAJAJAJ/PertemuanSET/nyobba.py"
{'upin', 'ipin', 'mail'}
○ PS C:\Users\jevon\OneDrive\Documents\Code JAJAJAAJAJAJ\PertemuanSET>

```

Difference

```

nyobba.py > ...
1  # bisa berbahasa english
2  english = {'desi', 'tono', 'evan', 'miko', 'takashi', 'chaewon'}
3  # bisa berbahasa korea
4  korean = {'chaewon', 'yeona', 'erika', 'miko'}
5  # siapa yang hanya bisa bahasa korea?
6  only_korean = korean - english
7  print(only_korean)
8  # siapa yang hanya bisa bahasa english?
9  only_english = english - korean
10 print(only_english)

```

```

● PS C:\Users\jevon\OneDrive\Documents\Code JAJAJAAJAJAJ\PertemuanSET> & C:/Users/jevon/OneDrive/Documents/Code JAJAJAAJAJAJ/PertemuanSET/nyobba.py"
{'yeona', 'erika'}
{'evan', 'takashi', 'desi', 'tono'}
○ PS C:\Users\jevon\OneDrive\Documents\Code JAJAJAAJAJAJ\PertemuanSET>

```

Symmetric Difference

```

nyobba.py > ...
1  english = {'desi', 'tono', 'evan', 'miko', 'takashi', 'chaewon'}
2  korean = {'chaewon', 'yeona', 'erika', 'miko'}
3  # hanya bisa bicara satu bahasa saja
4  one_language = english ^ korean
5  print(one_language)

```

```

● PS C:\Users\jevon\OneDrive\Documents\Code JAJAJAAJAJAJ\PertemuanSET> & C:/Users/jevon/OneDrive/Documents/Code JAJAJAAJAJAJ/PertemuanSET/nyobba.py"
{'desi', 'tono', 'evan', 'yeona', 'erika', 'takashi'}
○ PS C:\Users\jevon\OneDrive\Documents\Code JAJAJAAJAJAJ\PertemuanSET>

```

BAGIAN 2: LATIHAN MANDIRI (60%)

Mandiri 1

```
Mandiri1.py > ...
1
2 n = int(input('Masukkan jumlah kategori: '))
3 data_aplikasi = {}
4
5 for i in range(n):
6     nama_kategori = input('Masukkan nama kategori: ')
7     print('Masukkan 5 nama aplikasi di kategori', nama_kategori)
8
9     aplikasi = []
10    for j in range(5):
11        nama_aplikasi = input('Nama aplikasi: ')
12        aplikasi.append(nama_aplikasi)
13
14    data_aplikasi[nama_kategori] = aplikasi
15
16 print(data_aplikasi)
17
18 daftar_aplikasi_list = [set(aplikasi) for aplikasi in data_aplikasi.values()]
19
20
21 hasil_intersection = daftar_aplikasi_list[0]
22 for i in range(1, len(daftar_aplikasi_list)):
23     hasil_intersection = hasil_intersection.intersection(daftar_aplikasi_list[i])
24
25 print("Aplikasi yang ada di semua kategori:", hasil_intersection)
26
27 satukategori = daftar_aplikasi_list[0]
28 for i in range(1, len(daftar_aplikasi_list)):
29     satukategori = satukategori.symmetric_difference(daftar_aplikasi_list[i])
30 print("Aplikasi yang hanya ada di satu kategori:", satukategori-hasil_intersection)
31
32 duakategori = set()
33
34 for i in range(len(daftar_aplikasi_list) - 1):
35     hasil = (daftar_aplikasi_list[i].intersection(daftar_aplikasi_list[i+1])).difference(satukategori)
36     duakategori = duakategori.union(hasil)
37     hasil2 = (daftar_aplikasi_list[i].intersection(daftar_aplikasi_list[-1])).difference(satukategori)
38     duakategori = duakategori.union(hasil2)
39 print("Aplikasi yang muncul tepat di 2 kategori: ", duakategori)
40
41
```

Output

```
PS C:\Users\jevon\OneDrive\Documents\Code_JAJAJAJAJAJ\PertemuanSET> & C:/Users/jevon/AppData/Local/Microsoft/Windows/PowerShell/PowerShell.exe -Command python Mandiri1.py
Masukkan jumlah kategori: 3
Masukkan nama kategori: A
Masukkan 5 nama aplikasi di kategori A
Nama aplikasi: a
Nama aplikasi: b
Nama aplikasi: c
Nama aplikasi: d
Nama aplikasi: e
Masukkan nama kategori: B
Masukkan 5 nama aplikasi di kategori B
Nama aplikasi: a
Nama aplikasi: b
Nama aplikasi: f
Nama aplikasi: g
Nama aplikasi: h
Masukkan nama kategori: C
Masukkan 5 nama aplikasi di kategori C
Nama aplikasi: a
Nama aplikasi: i
Nama aplikasi: j
Nama aplikasi: k
Nama aplikasi: l
{'A': ['a', 'b', 'c', 'd', 'e'], 'B': ['a', 'b', 'f', 'g', 'h'], 'C': ['a', 'i', 'j', 'k', 'l']}
Aplikasi yang ada di semua kategori: {'a'}
Aplikasi yang hanya ada di satu kategori: {'j', 'c', 'f', 'h', 'l', 'e', 'k', 'g', 'i', 'd'}
Aplikasi yang muncul tepat di 2 kategori: {'b'}
PS C:\Users\jevon\OneDrive\Documents\Code_JAJAJAJAJAJ\PertemuanSET>
```

Source code diatas di ambil dari materi dan ditambahkan perulangan untuk menemukan aplikasi yang ada di 1 kategori saja dan aplikasi yang ada di 2 kategori

Mandiri 2

```
Mandiri2.py > ...
1  list1 = [1, 2, 3, 4, 5, 6, 6, 7, 8, 9]
2  set1 = set(list1)
3  print("List awal: ", list1)
4  print("Konversi Set: ", set1)
5
6  set2 = (1,2,3,4,5)
7  list2 = list(set2)
8  print("Set awal: ", set2)
9  print("Konversi List: ", list2)
10
11 tuple3 = (1,2,3,4,5,5,6,7,7)
12 set3 = set(tuple3)
13 print("Tuple awal: ", tuple3)
14 print("Konversi Set: ", set3)
15
16 set4 = {1,2,3,4,5,5,6,7,7}
17 tuple4 = tuple(set4)
18 print("Set awal: ", set4)
19 print("Konversi List: ", tuple4)
```

Output:

```
PS C:\Users\jevon\OneDrive\Documents\Code JAJAJAAJAJAJ\PertemuanSET> & C:/Users/jevon
List awal: [1, 2, 3, 4, 5, 6, 6, 7, 8, 9]
Konversi Set: {1, 2, 3, 4, 5, 6, 7, 8, 9}
Set awal: (1, 2, 3, 4, 5)
Konversi List: [1, 2, 3, 4, 5]
Tuple awal: (1, 2, 3, 4, 5, 5, 6, 7, 7)
Konversi Set: {1, 2, 3, 4, 5, 6, 7}
Set awal: {1, 2, 3, 4, 5, 6, 7}
Konversi List: (1, 2, 3, 4, 5, 6, 7)
PS C:\Users\jevon\OneDrive\Documents\Code JAJAJAAJAJAJ\PertemuanSET>
```

Source code diatas hanya melakukan perubahan dari bentuk awal ke bentuk akhir dengan menggunakan set(), list(), dan tuple()

Mandiri 3

```
Mandiri3.py > ...
1  try:
2      text1 = input("Nama Text 1 :")
3      text2 = input("Nama Text 2 :")
4      with open(text1, 'r') as file1, open(text2, 'r') as file2:
5          file1 = file1.read().lower().split()
6          file2 = file2.read().lower().split()
7          semua = set(file1)&set(file2)
8          for i in sorted(semua):
9              print(i)
10 except:
11     print("Text tidak ditemukan/Text salah(typo)!")
```

Output:

```
PS C:\Users\jevon\OneDrive\Documents\Code JAJAJAAJAJAJ\PertemuanSET> & C:/Use
Nama Text 1 :text1.txt
Nama Text 2 :text2.txt
halo
hari
malam
nama
saya
PS C:\Users\jevon\OneDrive\Documents\Code JAJAJAAJAJAJ\PertemuanSET> |
```

Source code diatas kedua file dibuka dan dilakukan intersection agar mendapatkan kata yang sama pada kedua text