



**Laporan Praktikum Algoritma & Pemrograman
Semester Genap 2024/2025**

SAYA MENYATAKAN BAHWA LAPORAN PRAKTIKUMINI 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 KULIAHINI.

NIM	<71220917>
Nama Lengkap	<Anggrayni Layuk Mangayun>
Minggu ke / Materi	08 / Pengolahan String dan Regular Expression

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
YOGYAKARTA
2025**

BAGIAN 1: MATERI MINGGUINI (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 Pengantar String

Adalah rangkain tipe data pada pemograman yang digunakan sebagai memory untuk menyimpan teks baik itu Panjang maupun pendek jadi string ini bisa menyimpan huruf ataupun karakter seperti emoji bisa juga disimpan dalam bentuk code ASCII. Perlu diketahui bahwah tidak semua bahasa pemograman mempunyai tipe data string contohnya : dalam bahasa C karena pada bahasa pemograman C String diimplementasikan sebagai array karakter, yaitu sekumpulan karakter yang tersimpan secara berurutan dalam memori.

Contoh bentuk string:

```
# 1. Deklarasi string sederhana
kalimat = 'selamat hari paskah hehehe'
print(kalimat)
# 2. String multi-baris
puisi = """musik adalah
anugerah dari
tuhan yesus kristus"""
print("\n" + puisi)
# 3. Menggunakan karakter khusus
emoji = "😊"
print("\n" + emoji)
# 4. Menggunakan ASCII
ascii_A = 65
ascii = chr(ascii_A)
print("Karakter dari ASCII 65:",ascii)
```

MATERI 2 Pengaksesan String dan Manipulasi String

bagaimana cara mengakses serta manipulasi?(mengubah,mengotakatik) string bisa menghapus,mengedit,menggabungkan perlu diketahui String bersifat immutable tidak bisa di ubah agar bisa di ubah harus disimpan dengan variable yang berbeda.

string simpan nya secara array dalam komputer Cara mengakses string adalah huruf pertama ditulis dengan indeks 0 .

```
#contoh akses string
nama_buah="banana"
print(nama_buah)
print(nama_buah[-2]) #output:n

nama = "anggi comel"
print(nama[0]) # Output: 'a' (menghitung indeks dimulai dari 0)
print(nama[-2]) # Output: 'e' (menghitung indeks negatif dari akhir)
print(nama[::-2]) # Output: agicml (mengambil setiap karakter kedua mulai dari indeks 0)
print(nama[4:6]) # Output: i (Mulai dari indeks ke-4 Berhenti sebelum karakter ke-6)
```

Bentuk lain nya yaitu manipulasi string

```
#gabungkan string
a='ukdw'
b='yogya'
print(a+b)

#berdasarkan code ASCII
a='A' # ASCII 65
b='a' # ASCII 97
print(a<b) #KNP hasilnya true karena huruf ascii A lebih kecil dari a

a='Anton' # Karakter pertama 'A' (65)
b='anton'# Karakter pertama 'a' (97) secara keseluruhan lebih kecil dari b
print(a>b)# knp hasil nya false
```

MATERI 3 Operator dan Metode String

Operator merupakan dasar dari tipe data untuk menggambarkan teks pada bahasa pemograman.ada berapa jenis operator namun berikut ini beberapa contoh nya:

OPERATOR Penggabungan

```
#OPERATOR penggabungan
a = "anggrayni"
b = "layuk"
```

```
nama_lengkap = a + " " + b # "anggrayni layuk"
```

OPERATOR in merupakan operator untuk mengecek apakah suatu kata atau teks itu ada pada string berikut ini contohnya dan penjelasan:

```
#Operator in (Membership Operator)
teks = "Anggi comel bgtt"
# Cek apakah "Python" ada dalam teks
print("Anggi" in teks) # Output: True
print("AnggY" in teks) # Output: FALSE karena kata tidak ada
# Cek apakah "cute" tidak ada dalam teks
print("Cute" not in teks) # Output: True
```

pada String juga dapat dilakukan perbandingan (comparison) contoh nya:

```
#berdasarkan code ASCII
a='A' # ASCII 65
b='a' # ASCII 97
print(a<b) # KNP hasilnya true karena huruf ascii A lebih kecil dari a

a='Anton' # Karakter pertama 'A' (65)
b='anton'# Karakter pertama 'a' (97) secara keseluruhan lebih kecil dari b
print(a>b)# knp hasil nya false
```

FUNGSI LEN untuk mengatahui panjang kata atau hruf ,spasi juga ikut di hitung len berguna untuk tau huruf nya berapa karena inputan nya dinamis.

```
kalimat = "Hey, patrick!"
print(len(kalimat)) # Output: 13

# Menggunakan dengan list
no = [1, 2, 3, 4, 5]
print(len(no)) # Output: 5
```

- **TRaversing** mengunjung setiap huruf untuk di ubah serta bisa memanipulasi perhuruf. Contohnya dibawah ini mengganti huruf menjadi angka 'a' → '4"e' → '3"s'→'2'
 - a="selmat pagi"
 - b="selamat malam"
 -

```

• print(b in a)
• for i in a: # Output: False
•     if i == 'a':
•         i='4'
•     elif i == 'e':
•         i='3'
•     elif i == 's':
•         i='2'
•     print(i,end="")
• # Output: 23lm4t p4giTrue

```

SLICE menampilkan bagian dari string . Slice dilakukan dengan menggunakan tanda kurung siku [] dan menentukan indeks awal dan akhir (opsional) untuk bagian String yang ingin diambil. Slice merupakan teknik yang sangat berguna dalam pemrograman python, memungkinkan kita untuk bekerja dengan teks dan karakter dalam string dengan cara yang paling lebih fleksibel dan efisien. contoh code nya dibawah ini dengan menggunakan indeks string [awal:akhir].

```

a="selmat pagi paskha"
b="selamat malam"
print(a[0:5])#output: selma
print(a[1:]) #output: elmat pagi paskha
print(a[::-1])# untuk balik kalimat,output:ahksap igap tamles

```

Berikut adalah beberapa method String yang sering digunakan:

Nama Method	Kegunaan	Penggunaan
capitalize()	untuk mengubah string menjadi huruf besar	string.capitalize()
count()	menghitung jumlah substring yang muncul dari sebuah string	string.count()
endswith()	mengetahui apakah suatu string diakhiri dengan string yang dininputkan	string.endswith()
startswith()	mengetahui apakah suatu string diawali dengan string yang dininputkan	string.startswith()
find()	mengembalikan indeks pertama string jika ditemukan string yang dicari	string.find()
islower() dan isupper()	mengembalikan True jika string adalah huruf kecil / huruf besar	string.islower() dan string.isupper()
isdigit()	mengembalikan True jika string adalah digit (angka)	string.isdigit()
strip()	menghapus semua whitespace yang ada di depan dan di akhir string	string.strip()
split()	memecah string menjadi token-token berdasarkan pemisah, misalnya berdasarkan spasi	string.split()

MATERI 4 Parsing String

PARSING yaitu string jadi menulusuri string untuk mendapat sesuatu atau mengubah sesuatu atau string yang di inginkan. Contoh code berikut tentang mencari tanggal dd/mm/yyyy pada suatu kalimat:

```
kalimat="saya berulang tahun 20-03-2005 dan jimin ulang tahun 18-08-2005 kami pacaran"
hasil_spilit = kalimat.split(" ")
print(hasil_spilit)

for i in hasil_spilit: #code mencari tanggal dd/mm/yyyy
    if i[0].isdigit():#menegcek apakah hruf depan nya angka atau bukan
        print(i)
```

output:

```
['saya', 'berulang', 'tahun', '20-03-2005', 'dan', 'jimin', 'ulang', 'tahun', '18-08-2005', 'kami', 'pacaran']
20-03-2005
18-08-2005
```

MATERI 5 Pengantar Regex

PENGANTAR REGEX (pola regular expression)berbentuk kalimat atau kata pola yg bisa kita cari dalam sebuah string (sanagt berguna kalau teks nya panjang)

MATERI 6 Meta Character, Escaped Character, Set of Character, dan Fungsi Regex pada Libary Python

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.

LINK: https://github.com/anggicute44/71220917_laporan8_pralpro.git

SOAL 1

Latihan 8.5 Anda diminta untuk mencari seluruh teks yang berupa tanggal dengan format YYYY-MM-DD dan kemudian seluruh tanggal tersebut diambil dan ditampilkan kembali dalam format DD-MM-YYYY ditambah dengan perhitungan selisih dengan tanggal sekarang dalam hari.

Contoh:

Pada tanggal 1945-08-17 Indonesia merdeka. Indonesia memiliki beberapa pahlawan nasional, seperti Pangeran Diponegoro (TL: 1785-11-11), Pattimura (TL: 1783-06-08) dan Ki Hajar Dewantara (1889-05-02).

Hasil:

1945-08-17 00:00:00 selisih 27209 hari
1785-11-11 00:00:00 selisih 85561 hari
1783-06-08 00:00:00 selisih 86448 hari
1889-05-02 00:00:00 selisih 47769 hari

Code:

```
from datetime import datetime # untuk manipulasi tanggal dan waktu

def process_dates(text):

    import re # Mengimpor modul regex

    # Pola regex untuk menemukan tanggal YYYY-MM-DD
    polaTgl = r'\b\d{4}-\d{2}-\d{2}\b'
    tglKetemu = re.findall(polaTgl, text) # untuk mencocok tgl dengan pola dalam teks

    # untuk Mendapatkan tanggal hari ini
    harii = datetime.today()

    for i in tglKetemu:
        try:
            # untuk mengubah string tanggal menjadi objek datetime
```

```
tgl_obj = datetime.strptime(i, '%Y-%m-%d')

print(f"{i} 00:00:00 selisih {(harii - tgl_obj).days} hari") #Menghitung selisih hari
except ValueError:
    continue

# Contoh penggunaan
kalimatsoal = """
Pada tanggal 1945-08-17 Indonesia merdeka. Indonesia memiliki beberapa pahlawan
nasional, seperti Pangeran Diponegoro (TL: 1785-11-11), Pattimura (TL: 1783-06-08) dan Ki
Hajar Dewantara (1889-05-02).
"""

process_dates(kalimatsoal)

penjelasan: Pertama, kita menggunakan regex untuk mencari semua pola tanggal format
YYYY-MM-DD dalam teks. Kedua, setiap tanggal yang ditemukan diubah dari string menjadi
objek datetime agar bisa diproses. Terakhir, kode menghitung selisih hari antara tanggal
tersebut dengan hari ini, lalu menampilkan hasilnya dalam format yang diminta. Dengan
pendekatan ini, kode mampu mengekstrak tanggal secara akurat dari berbagai posisi
dalam teks dan menghitung selisih waktu dengan presisi.
```

Output:

```
D:\laporan alpro\code_laporan_8>python -u "d:\laporan alpro\code_laporan_8\8.1.py"
1945-08-17 00:00:00 selisih 29108 hari
1785-11-11 00:00:00 selisih 87460 hari
1783-06-08 00:00:00 selisih 88347 hari
1889-05-02 00:00:00 selisih 49668 hari
```

SOAL 2

Latihan 8.6 Anda diminta untuk mencari seluruh teks yang berupa email dan kemudian ambil semua username dari email tersebut untuk digenerate password random 8 karakter yang terdiri dari angka dan huruf.

Contoh:

Berikut adalah daftar email dan nama pengguna dari mailing list:
anton@mail.com dimiliki oleh antonius
budi@gmail.co.id dimiliki oleh budi anwari
slamet@getnada.com dimiliki oleh slamet slumut
matahari@tokopedia.com dimiliki oleh toko matahari

Hasil:

anton@mail.com username: anton , password: 8u78A2UD
budi@gmail.co.id username: budi , password: bdP066Ld
slamet@getnada.com username: slamet , password: Ab1FiHXb
matahari@tokopedia.com username: matahari , password: 5KYyaP6



Code:

```
import re
import random #Untuk generate karakter acak (password)
import string

def mencariemail_passwords(text):

    # Pola regex untuk mencocokkan email
    pola_Email = r"\b[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Z|a-z]{2,}\b"
    emails = re.findall(pola_Email, text) # untuk mengembalikan semua email yang cocok dengan pola.

    for email in emails:
        # Untuk Ekstrak username (bagian sebelum @)
        username = email.split('@')[0]

        # Generate password random 8 karakter (huruf besar, kecil, dan angka)
        karakter = string.ascii_letters + string.digits # Gabung huruf dan angka
        password = ''.join(random.choice(karakter) for _ in range(8))# untuk gabungkan Password 8 karakter

        print(f"{email} username: {username}, password: {password}")

# Contoh sesuai soal
kalimat = """"
```

Berikut adalah daftar email dan nama pengguna dari mailing list:

anton@mail.com dimiliki oleh antonius

budi@gmail.co.id dimiliki oleh budi anwari

slamet@getnada.com dimiliki oleh slamet slumut

matahari@tokopedia.com dimiliki oleh toko matahari

.....

mencariemail_passwords(kalimat)

penjelasan:

Pertama, program ini memakai tiga alat bantu: re untuk mencari pola dalam tulisan, random untuk membuat karakter secara acak, dan string untuk menyediakan kumpulan huruf dan angka. kemudian cari pola untuk alamat email dalam sebuah teks dengan menggunakan pola khusus. Pola digunakan: = `r'\b[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Z|a-z]{2,}\b'`

- Pola ini mencocokkan format email standar:

- `[A-Za-z0-9._%+-]+` : Username (boleh mengandung huruf, angka, titik, dll).
- `@[A-Za-z0-9.-]+` : Domain (setelah `@`).
- `\.[A-Z|a-z]{2,}` : Ekstensi (`.com`, `.co.id`, dll).

Setiap email yang ditemukan akan dipisahkan menjadi dua bagian berdasarkan simbol '@'. Bagian sebelum tanda '@' akan diambil sebagai username. Setelah mendapatkan username, langkah selanjutnya adalah membuat password acak. Untuk setiap username, program akan menghasilkan password dengan panjang 8 karakter, yang merupakan kombinasi huruf besar, huruf kecil, dan angka. Karakter-karakter ini dipilih secara acak dari kumpulan huruf dan angka yang tersedia.

Output:

```
anton@mail.com username: anton, password: mwOhT0mk
budi@gmail.co.id username: budi, password: ACwnZB4T
slamet@getnada.com username: slamet, password: wCBZ9Jze
matahari@tokopedia.com username: matahari, password: vuUJYywI
```