# Tugas Struktur Data

### Praktek 1:

```
# impor library numpy
import numpy as np

# membuat array dengan numpy
nilai_siswa = np.array([85, 55, 40, 90])

# akses data pada array
print(nilai_siswa[3])
```

# Hasilnya:

```
Share $ Command Line Arguments

90

** Process exited - Return Code: 0 **

Press Enter to exit terminal
```

# Penjelasan:

Baris -1: Mengimpor library NumPy dengan alias np.

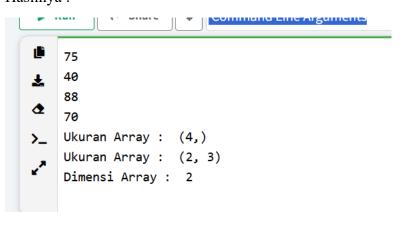
Baris -2 : Membuat array NumPy berisi nilai-nilai siswa

Baris -3: Menampilkan nilai pada indeks ke-3, yaitu 90.

#### Praktek 2:

```
# impor libaray numpy
import numpy as np
# membuat array dengan numpy
nilai_siswa_1 = np.array([75, 65, 45, 80])
nilai_siswa_2 = np.array([[85, 55, 40], [50, 40, 99]])
# cara akses elemen array
print(nilai_siswa_1[0])
print(nilai_siswa_2[1][1])
# mengubah nilai elemen array
nilai_siswa_1[0] = 88
nilai_siswa_2[1][1] = 70
# cek perubahannya dengan akses elemen array
print(nilai_siswa_1[0])
print(nilai_siswa_2[1][1])
# Cek ukuran dan dimensi array
print("Ukuran Array : ", nilai_siswa_1.shape)
print("Ukuran Array : ", nilai_siswa_2.shape)
print("Dimensi Array : ", nilai_siswa_2.ndim)
```

# Hasilnya:



# Penjelasannya:

```
Baris -1: Impor library NumPy.
```

Baris -2: Membuat array 1 dimensi (nilai siswa 1).

Baris -3: Membuat array 2 dimensi (nilai siswa 2).

Baris -4 : Menampilkan elemen pertama array 1D  $\rightarrow$  75.

Baris -5 : Menampilkan elemen baris ke-2, kolom ke-2  $\rightarrow$  40.

Baris -6: Mengubah elemen pertama nilai siswa 1 menjadi 88.

Baris -7 : Mengubah elemen baris ke-2, kolom ke-2 nilai\_siswa\_2 menjadi 70.

Baris -8 : Menampilkan elemen pertama nilai siswa 1 setelah diubah  $\rightarrow$  88.

Baris -9: Menampilkan elemen nilai\_siswa\_2[1][1] setelah diubah  $\rightarrow$  70.

Baris -10: Menampilkan ukuran (jumlah elemen) array 1D  $\rightarrow$  (4,).

Baris -11 : Menampilkan ukuran array 2D  $\rightarrow$  (2, 3) (2 baris, 3 kolom).

Baris -12 : Menampilkan jumlah dimensi array  $\rightarrow$  2.

### Praktek 3:

```
# impor library numpy
import numpy as np

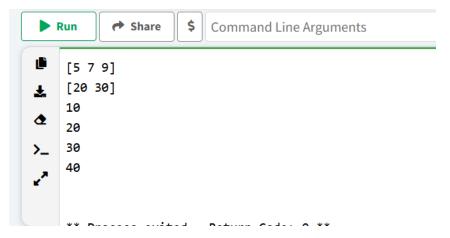
# membuat array
a = np.array([1, 2, 3])
b = np.array([4, 5, 6])

# menggunakan operasi penjumlahan pada 2 array
print(a + b)  # array([5, 7, 9])

# Indexing dan Slicing pada Array
arr = np.array([10, 20, 30, 40])
print(arr[1:3])  # array([20, 30])

# iterasi pada array
for x in arr:
    print(x)
```

Hasilnya:



# Penjelasannya:

Baris -1: Mengimpor library NumPy.

Baris -2 &-3: Membuat dua array 1 dimensi a dan b.

Baris -4 : Menjumlahkan dua array secara elemen-per-elemen  $\rightarrow$  [1+4, 2+5, 3+6].

Baris -5: Membuat array arr dengan empat elemen.

Baris -6: Menampilkan elemen dari indeks 1 sampai sebelum  $3 \rightarrow [20, 30]$ .

Baris -7 & -8: Melakukan iterasi (perulangan) dan mencetak setiap elemen dalam array arr.

### Praktek 4:

```
# membuat array
arr = [1, 2, 3, 4, 5]

# Linear Traversal ke tiap elemen arr
print("Linear Traversal: ", end=" ")
for i in arr:
    print(i, end=" ")
print()
```

Hasilnya:

```
Linear Traversal: 1 2 3 4 5

** Process exited - Return Code: 0 **

Press Enter to exit terminal
```

# Penjelasannya;

Baris -1: Membuat list arr berisi angka 1 sampai 5.

Baris -2: Mencetak teks "Linear Traversal: " tanpa pindah baris.

Baris -3&-4 : Melakukan perulangan untuk mencetak setiap elemen arr dalam satu baris, dipisah spasi.

Baris -5: Pindah baris setelah semua elemen dicetak.

### Praktek 5:

```
# membuat array
arr = [1, 2, 3, 4, 5]

# Reverse Traversal dari elemen akhir
print("Reverse Traversal: ", end="")
for i in range(len(arr) - 1, -1, -1):
    print(arr[i], end=" ")
print()
```

Hasilnya:



# Penjelasannya:

Baris -1 : Membuat list arr berisi angka 1 sampai 5.

Baris -2: Mencetak teks "Reverse Traversal: " tanpa pindah baris.

Baris -3: Melakukan perulangan dari indeks terakhir ke pertama:

- $len(arr) 1 \rightarrow indeks terakhir (4)$
- $-1 \rightarrow$  batas akhir (tidak termasuk)
- $-1 \rightarrow langkah mundur$

Baris -4: Mencetak elemen arr secara terbalik (dari belakang ke depan), dipisah spasi.

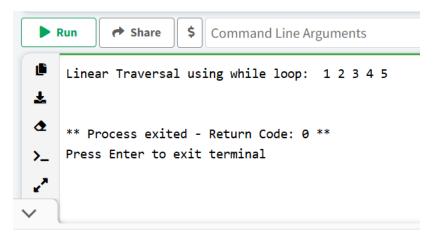
Baris -5: Pindah baris setelah selesai mencetak semua elemen.

#### Praktek 7:

```
# membuat array
arr = [1, 2, 3, 4, 5]

# mendeklarasikan nilai awal
n = len(arr)
i = 0

print("Linear Traversal using while loop: ", end=" ")
# Linear Traversal dengan while
while i < n:
    print(arr[i], end=" ")
    i += 1
print()</pre>
```



### Penjelasannya:

- Baris -1: Membuat list arr berisi angka 1 sampai 5.
- Baris -2: Menyimpan panjang list arr ke dalam variabel n (hasilnya 5).
- Baris -3: Inisialisasi variabel indeks i dengan nilai awal 0.
- Baris -4: Mencetak teks "Linear Traversal using while loop: " tanpa pindah baris.
- Baris -5: Perulangan selama nilai i kurang dari panjang array (n).
- Baris -6: Mencetak elemen ke-i dari array, dipisah spasi.
- Baris -7: Menambah nilai i satu per satu (untuk lanjut ke indeks berikutnya).
- Baris -8: Pindah baris setelah semua elemen dicetak.

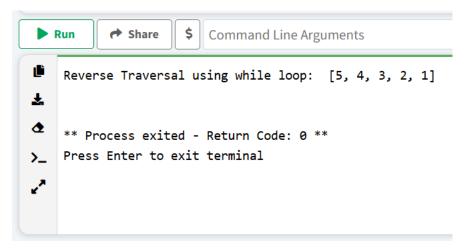
### Praktek 8:

```
# membuat array
arr = [1, 2, 3, 4, 5]

# mendeklarasikan nilai awal
start = 0
end = len(arr) - 1

print("Reverse Traversal using while loop: ", end=" ")
# Reverse Traversal dengan while
while start < end:

arr[start], arr[end] = arr[end], arr[start]
    start += 1
    end -= 1
print(arr)</pre>
```



### Penjelasannya:

Baris -1: Membuat list arr berisi angka 1 sampai 5.

Baris -2&-3: Inisialisasi indeks start (awal) dan end (akhir) untuk proses pembalikan.

Baris -4: Mencetak teks "Reverse Traversal using while loop: " tanpa pindah baris.

Baris -5: Melakukan perulangan selama start lebih kecil dari end.

Baris -6: Menukar elemen di posisi start dengan elemen di posisi end.

Baris -7&-8: Maju satu langkah dari depan (start) dan mundur satu langkah dari belakang (end).

Baris -9: Mencetak array yang sudah dibalik.

Output akhir: [5, 4, 3, 2, 1]

#### Praktek 9:

```
# membuat array
arr = [12, 16, 20, 40, 50, 70]

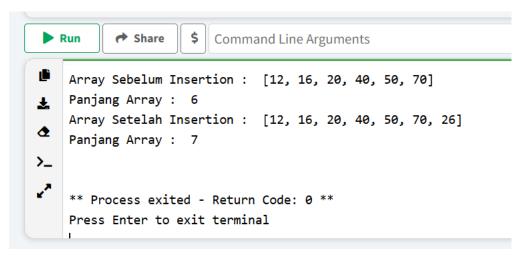
# cetak arr sebelum penyisipan
print("Array Sebelum Insertion : ", arr)

# cetak panjang array sebelum penyisipan
print("Panjang Array : ", len(arr))

# menyisipkan array di akhir elemen menggunakan .append()
arr.append(26)

# cetak arr setelah penyisipan
print("Array Setelah Insertion : ", arr)

# cetak panjang array setelah penyisipan
print("Panjang Array : ", len(arr))
```



### Penjelasannya:

- Baris -1 : Membuat list arr berisi beberapa angka.
- Baris -2: Menampilkan array sebelum elemen baru ditambahkan.
- Baris -3: Menampilkan jumlah elemen array sebelum penambahan.
- $Baris\ \mbox{-}4$  : Menambahkan angka 26 ke akhir array menggunakan fungsi append().
- Baris -5: Menampilkan array setelah elemen 26 disisipkan.
- Baris -6: Menampilkan panjang array setelah penambahan elemen.

### Praktek 10:

```
# membuat array
arr = [12, 16, 20, 40, 50, 70]

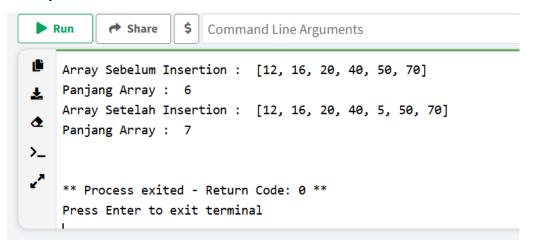
# cetak arr sebelum penyisipan
print("Array Sebelum Insertion : ", arr)

# cetak panjang array sebelum penyisipan
print("Panjang Array : ", len(arr))

# menyisipkan array pada tengah elemen menggunakan .insert(pos arr.insert(4, 5)

# cetak arr setelah penyisipan
print("Array Setelah Insertion : ", arr)

# cetak panjang array setelah penyisipan
print("Panjang Array : ", len(arr))
```



### Penjelasannya:

- Baris -1: Membuat list arr berisi angka.
- Baris -2: Menampilkan array sebelum penambahan elemen.
- Baris -3: Menampilkan jumlah elemen array sebelum penyisipan (hasilnya 6).
- Baris -4: Menyisipkan angka 5 pada indeks ke-4 (sebelum elemen 50).
- Baris-5: Menampilkan array setelah elemen 5 disisipkan.
- Baris -6: Menampilkan panjang array setelah penyisipan (hasilnya 7).