

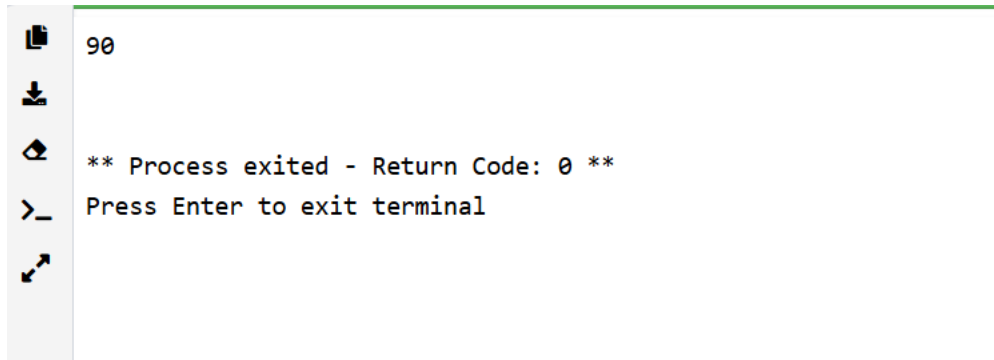
Praktek 1:

```
# 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:



```
90

** Process exited - Return Code: 0 **
Press Enter to exit terminal
```

Penjelasan:

Baris ke-1: Baris ini mengimpor library numpy, yaitu library Python yang digunakan untuk operasi numerik, Khususnya array dan matriks. Kata as np digunakan untuk menyingkat penulisan numpy menjadi np, sehingga lebih praktis digunakan.

Baris ke-2: Baris ini membuat sebuah array numpy yang berisi data nilai siswa: 85, 55, 40, dan 90. Fungsi np.array([...]) digunakan untuk membuat array dari daftar (list) Python.

Baris ke-3: Baris ini mencetak elemen ke-4 dari array nilai_siswa, yaitu 90. Perlu diingat bahwa indeks dimulai dari 0, sehingga:

- o Nilai_siswa[0] adalah 85
- o Nilai_siswa[1] adalah 55
- o Nilai_siswa[2] adalah 40
- o Nilai_siswa[3] adalah 90

Praktek 2:

```
# impor library 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:

	75
	40
	88
	70
	Ukuran Array : (4,)
	Ukuran Array : (2, 3)
	Dimensi Array : 2

Penjelasan:

Baris ke-1: Mengimpor library numpy dan disingkat menjadi np agar lebih ringkas saat digunakan.

Baris ke-2: Membuat array dengan numpy.

- nilai_siswa_1: array 1 dimensi yang berisi nilai-nilai siswa.
- nilai_siswa_2: array 2 dimensi (matriks 2 x 3), yang terdiri dari 2 baris dan 3 kolom.

Baris ke-3: Cara akses elemen array

- nilai_siswa_1[0]: mengakses elemen pertama dari array 1 dimensi (75).
- nilai_siswa_2[1][1]: mengakses elemen baris ke-2, kolom ke-2 dari array 2 dimensi (40 sebelum diubah)

Baris ke-4: Mengubah Nilai elemen array

- mengubah nilai pertama di nilai_siswa_1 menjadi 88.
- Mengubah nilai pada baris ke-2, kolom ke-2 di nilai_siswa_2 menjadi 70.

Baris ke-5: Mengecek perubahannya dengan akses elemen array

- Menampilkan nilai yang sudah diperbarui 88 dan 70.

Baris ke-6: Mengecek ukuran dan dimensi array

- shape: menunjukkan ukuran/tata letak array.
 - o nilai_siswa_1. shape = (4,) artinya 1 dimensi dengan 4 elemen.
 - o nilai_siswa_2. shape = (2, 3) artinya 2 baris dan 3 kolom.
- ndim: menunjukkan jumlah dimensi array.
 - o nilai_siswa_2. Ndim = 2 karena array 2 dimensi.

Praktek 3:

```
# impor library numpy
import numpy as np

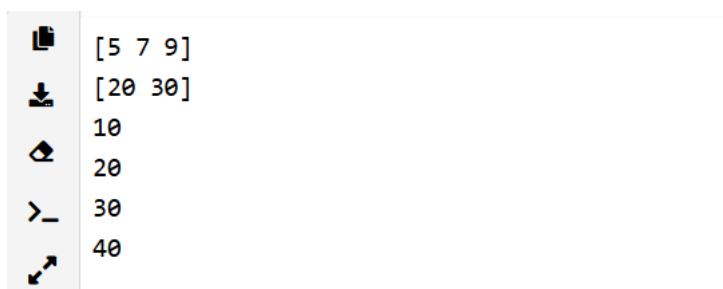
# membuat array
a = np.array([1, 2, 3])
b = np.array([4, 5, 6])

# 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:



```
[5 7 9]
[20 30]
10
20
30
40
```

Penjelasan:

Baris ke-1: Komentar bahwa baris selanjutnya akan mengimpor library numpy.

Baris ke-2: Mengimpor library numpy dan menyingkatnya menjadi np agar lebih ringkas saat dipakai.

Baris ke-3: Komentar bahwa bagian ini akan membuat array menggunakan numpy.

Baris ke-4: Membuat array a berisi elemen [1,2,3].

Baris ke-5: Membuat array b berisi elemen [4,5,6].

Baris ke-6: Komentar bahwa bagian ini melakukan operasi penjumlahan antar array.

Baris ke-7: Menjumlahkan elemen-elemen array a dan b satu per satu:

- $1 + 4 = 5$
- $2 + 5 = 7$
- $3 + 6 = 9$

Baris ke-8: Komentar bahwa bagian ini akan menunjukkan cara mengambil Sebagian elemen dari array (slicing).

Bagian ke-9: Membuat array arr berisi elemen [10,20,30,40].

Bagian ke-10: Mengambil elemen dari indeks ke-1 sampai sebelum indeks ke-3 (slicing)

- Indeks ke-1 = 20
- Indeks ke-2 = 30

Bagian ke-11: Komentar bahwa bagian ini akan melakukan perulangan pada elemen array.

Bagian ke-12: Melakukan iterasi (perulangan) untuk setiap elemen x di dalam array arr.

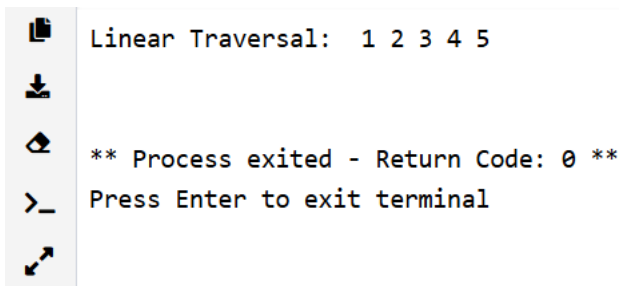
Bagian ke-13: Menampilkan setiap nilai x (elemen array).

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
```

Penjelasan:

Baris ke-1: Komentar yang menjelaskan bahwa baris berikutnya akan membuat array atau daftar.

Baris ke-2: Membuat array (list) bernama arr yang berisi lima elemen angka, yaitu 1 sampai 5.

Baris ke-3: Komentar yang menjelaskan bahwa bagian ini akan melakukan penelusuran (traversal) secara linear ke setiap elemen array.

Baris ke-4: Mencetak teks "Linear Traversal:" ke layar tanpa berpindah ke baris baru, agar elemen-elemen array dapat dicetak di baris yang sama.

Baris ke-5: Memulai perulangan for untuk menelusuri setiap elemen di dalam array arr.

Baris ke-6: Mencetak elemen saat ini (i) tanpa berpindah baris, dan memberi spasi setelah setiap angka.

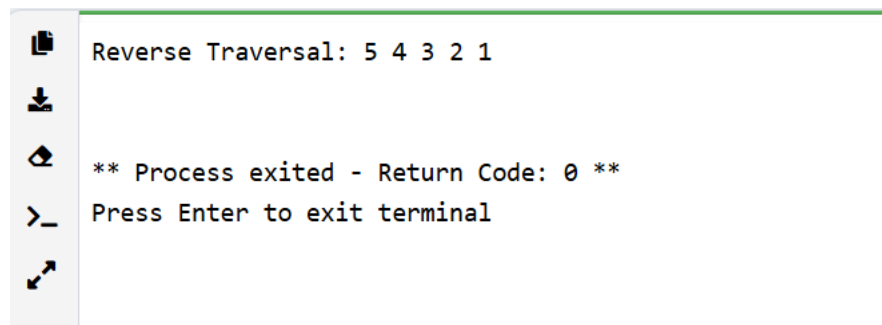
Baris ke-7: Mencetak baris baru setelah semua elemen array ditampilkan, agar tampilan rapi dan tidak menempel ke output berikutnya.

Praktik 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:



```
Reverse Traversal: 5 4 3 2 1

** Process exited - Return Code: 0 **
Press Enter to exit terminal
```

Penjelasan:

Baris ke-1: Komentar yang menjelaskan bahwa baris selanjutnya akan membuat array atau list.

Baris ke-2: Membuat array (list) bernama arr yang berisi lima elemen: 1, 2, 3, 4, dan 5.

Baris ke-3: Komentar yang menjelaskan bahwa bagian ini akan menelusuri (traversal) elemen array dari belakang ke depan (reverse).

Baris ke-4: Mencetak teks "Reverse Traversal:" ke layar tanpa ganti baris, supaya elemen-elemen bisa muncul di baris yang sama.

Baris ke-5: Melakukan perulangan for untuk mengambil indeks dari elemen terakhir (len(arr) - 1) sampai indeks pertama (0) secara mundur (-1 sebagai langkah).

Baris ke-6: Mencetak elemen array berdasarkan indeks i, dengan spasi di antaranya dan tetap di baris yang sama.

Baris ke-7: Mencetak baris kosong setelah semua elemen ditampilkan agar output rapi dan tidak menempel ke output berikutnya.

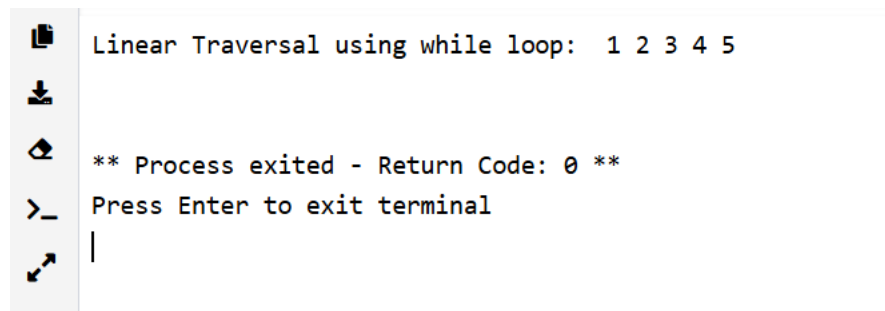
Praktek 7:

```
# 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()
```

Hasilnya:



```
Linear Traversal using while loop: 1 2 3 4 5

** Process exited - Return Code: 0 **
Press Enter to exit terminal
```

Penjelasan:

Baris ke-1: Komentar yang menjelaskan bahwa baris berikutnya akan membuat array (list).

Baris ke-2: Membuat array (list) bernama arr yang berisi lima elemen: 1, 2, 3, 4, dan 5.

Baris ke-3: Komentar yang menjelaskan bahwa nilai awal akan dideklarasikan sebelum melakukan perulangan.

Baris ke-4: Menyimpan panjang array arr ke dalam variabel n. Ini akan digunakan sebagai batas akhir perulangan.

Baris ke-5: Menginisialisasi variabel penghitung i dengan nilai 0. Ini sebagai indeks awal untuk mengakses elemen array.

Baris ke-6: Mencetak teks "Linear Traversal using while loop:" ke layar tanpa ganti baris, agar hasil traversal muncul di baris yang sama.

Baris ke-7: Komentar yang menjelaskan bahwa bagian ini menggunakan while loop untuk traversal array.

Baris ke-8: Memulai perulangan while selama i masih lebih kecil dari n (panjang array).

Baris ke-9: Mencetak elemen array pada indeks ke-i di baris yang sama, dipisahkan oleh spasi.

Baris ke-10: Menambahkan nilai i sebanyak 1 untuk melanjutkan ke elemen berikutnya.

Baris ke-11: Mencetak baris kosong setelah traversal selesai, supaya output rapi dan tidak menempel pada baris selanjutnya.

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)
```

Hasilnya:

```
Reverse Traversal using while loop: [5, 4, 3, 2, 1]

** Process exited - Return Code: 0 **
Press Enter to exit terminal
|
```

Penjelasan:

Baris ke-1: Komentar yang menjelaskan bahwa baris berikutnya membuat array (list).

Baris ke-2: Membuat array bernama arr yang berisi lima elemen: 1, 2, 3, 4, dan 5.

Baris ke-3: Komentar yang menunjukkan bahwa kita akan mendeklarasikan nilai awal sebelum perulangan.

Baris ke-4: Membuat variabel start dengan nilai 0 sebagai indeks awal.

Baris ke-5: Membuat variabel end dengan nilai indeks terakhir array (panjang array dikurangi 1).

Baris ke-6: Mencetak teks "Reverse Traversal using while loop:" di layar tanpa ganti baris, agar output berada di baris yang sama.

Baris ke-7: Komentar yang menjelaskan bahwa perulangan dilakukan secara mundur menggunakan while.

Baris ke-8: Memulai perulangan while selama nilai start masih lebih kecil dari end.

Baris ke-9: Menukar elemen array pada indeks start dan end. Ini akan membalik posisi elemen dari ujung ke tengah.

Baris ke-10: Menambahkan nilai start sebanyak 1 untuk maju ke indeks berikutnya dari depan.

Baris ke-11: Mengurangi nilai end sebanyak 1 untuk mundur ke indeks sebelumnya dari belakang.

Baris ke-12: Mencetak array setelah proses pembalikan selesai.

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))
```

Hasilnya:


```
Array Sebelum Insertion : [12, 16, 20, 40, 50, 70]
Panjang Array : 6
Array Setelah Insertion : [12, 16, 20, 40, 50, 70, 26]
Panjang Array : 7

** Process exited - Return Code: 0 **
Press Enter to exit terminal
```

Penjelasan:

Baris ke-1: Komentar yang menjelaskan bahwa baris berikutnya membuat array.

Baris ke-2: Membuat array bernama arr yang berisi enam elemen: 12, 16, 20, 40, 50, dan 70.

Baris ke-3: Komentar yang menjelaskan bahwa baris selanjutnya akan mencetak array sebelum dilakukan penyisipan elemen.

Baris ke-4: Mencetak array arr sebelum elemen baru disisipkan.

Baris ke-5: Komentar yang menyatakan bahwa baris berikut akan mencetak panjang array sebelum penyisipan.

Baris ke-6: Mencetak jumlah elemen dalam array arr sebelum penyisipan.

Baris ke-7: Komentar yang menjelaskan bahwa penyisipan akan dilakukan di akhir array menggunakan metode .append().

Baris ke-8: Menyisipkan elemen 26 ke akhir array menggunakan metode .append().

Baris ke-9: Komentar yang menyatakan bahwa array akan dicetak setelah elemen disisipkan.

Baris ke-10: Mencetak array arr setelah elemen 26 ditambahkan.

Baris ke-11: Komentar yang menyatakan bahwa panjang array akan dicetak setelah penyisipan.

Baris ke-12: Mencetak jumlah elemen array setelah elemen baru ditambahkan.

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))

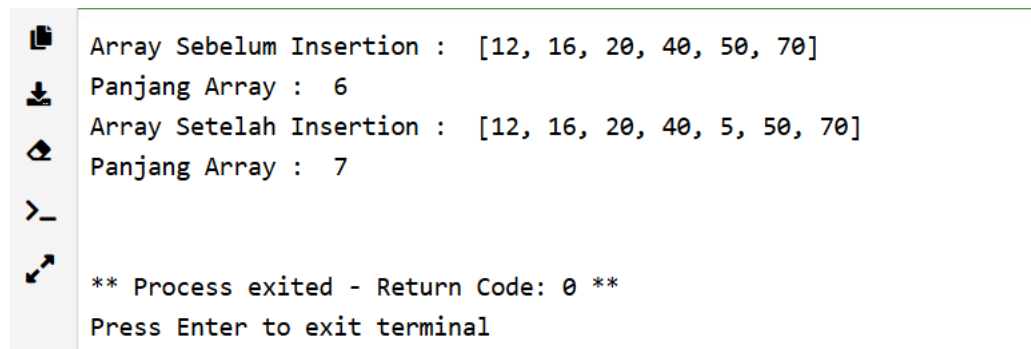
# menyisipkan array pada tengah elemen menggunakan .insert(pos, x)
arr.insert(4, 5)

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

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

```

Hasilnya:



```

Array Sebelum Insertion : [12, 16, 20, 40, 50, 70]
Panjang Array : 6
Array Setelah Insertion : [12, 16, 20, 40, 5, 50, 70]
Panjang Array : 7

** Process exited - Return Code: 0 **
Press Enter to exit terminal

```

Penjelasan:

Baris ke-1: Komentar yang menjelaskan bahwa array (list) akan dibuat.

Baris ke-2: Membuat array bernama arr dengan isi: 12, 16, 20, 40, 50, 70.

Baris ke-3: Komentar bahwa array akan dicetak sebelum elemen baru disisipkan.

Baris ke-4: Mencetak array sebelum proses penyisipan dilakukan.

Baris ke-5: Komentar bahwa panjang array akan dicetak sebelum penyisipan.

Baris ke-6: Mencetak jumlah elemen array sebelum disisipkan elemen baru.

Baris ke-7: Komentar yang menjelaskan bahwa elemen akan disisipkan di tengah array menggunakan metode .insert(posisi, nilai).

Baris ke-8: Menyisipkan angka 5 ke indeks ke-4 (yaitu sebelum angka 50). Jadi, 5 akan berada di antara 40 dan 50.

Baris ke-9: Komentar bahwa array akan dicetak setelah elemen berhasil disisipkan.

Baris ke-10: Mencetak array setelah elemen 5 disisipkan.

Baris ke-11: Komentar bahwa panjang array akan dicek kembali setelah penyisipan.

Baris ke-12: Mencetak panjang array setelah elemen 5 ditambahkan, kini bertambah menjadi 7 elemen.