

NAMA :IMAM ARDI PERDANA

NIM :24241020

KELAS :PTI A

MATKUL :STRUKTUR DATA

TUGAS double linked-list

Pratek

```
b tugas 2 > double linked list.py > ...
1  class Node:
2      def __init__(self, data):
3          self.data = data
4          self.prev = None
5          self.next = None
6
7  class DoubleLinkedList:
8      def __init__(self):
9          self.head = None
10
11     # Menambahkan node di akhir list
12     def append(self, data):
13         new_node = Node(data)
14         if self.head is None:
15             self.head = new_node
16             return
17         curr = self.head
18         while curr.next:
19             curr = curr.next
20         curr.next = new_node
21         new_node.prev = curr
22
23     # Menampilkan isi list
24     def display(self):
25         curr = self.head
26         while curr:
27             print(curr.data, end=' <-> ')
28             curr = curr.next
29         print("None")
```

```

31     # 1. Menghapus node awal
32     def delete_from_beginning(self):
33         if self.head is None:
34             print("List kosong.")
35             return
36         print(f"Menghapus node awal: {self.head.data}")
37         self.head = self.head.next
38         if self.head:
39             self.head.prev = None
40
41     # 2. Menghapus node akhir
42     def delete_from_end(self):
43         if self.head is None:
44             print("List kosong.")
45             return
46         curr = self.head
47         if curr.next is None:
48             print(f"Menghapus node akhir (tunggal): {curr.data}")
49             self.head = None
50             return
51         while curr.next:
52             curr = curr.next
53         print(f"Menghapus node akhir: {curr.data}")
54         curr.prev.next = None

```

```

56     # 3. Menghapus node berdasarkan nilai data
57     def delete_by_value(self, value):
58         curr = self.head
59         while curr:
60             if curr.data == value:
61                 print(f"Menghapus node dengan nilai: {value}")
62                 # Node di awal
63                 if curr.prev is None:
64                     self.head = curr.next
65                     if self.head:
66                         self.head.prev = None
67                 # Node di tengah atau akhir
68                 else:
69                     curr.prev.next = curr.next
70                     if curr.next:
71                         curr.next.prev = curr.prev
72                 return
73             curr = curr.next
74         print(f"Node dengan nilai {value} tidak ditemukan.")

```

```

76  # === Contoh penggunaan ===
77
78  dll = DoubleLinkedList()
79  dll.append(10)
80  dll.append(20)
81  dll.append(30)
82  dll.append(40)
83
84  print("Isi awal linked list:")
85  dll.display()
86
87  dll.delete_from_beginning()
88  dll.display()
89
90  dll.delete_from_end()
91  dll.display()
92
93  dll.delete_by_value(20)
94  dll.display()
95
96  dll.delete_by_value(99) # Coba hapus data yang tidak ada

```

Hasil Ouputnya

```

PS E:\algoritma\Tugas-semester-2\Modul 2> & C:/Users/USER/AppData/Local/Programs/Python/Python313/python.exe
Modul 2/b tugas 2/double linked list.py"
Isi awal linked list:
10 <-> 20 <-> 30 <-> 40 <-> None
Menghapus node awal: 10
20 <-> 30 <-> 40 <-> None
Menghapus node akhir: 40
20 <-> 30 <-> None
Menghapus node dengan nilai: 20
30 <-> None
Node dengan nilai 99 tidak ditemukan.
PS E:\algoritma\Tugas-semester-2\Modul 2>

```

Penjelasan Program Double Linked List (Bahasa Sederhana)

Bagian 1: Kelas Node

- **Baris 1–5**

Kita membuat sebuah class bernama Node.

Ini seperti satu kotak yang menyimpan data.

Setiap kotak punya tiga bagian:

- **data:** isi kotaknya (angka atau nilai lain),
- **prev:** menunjuk ke kotak sebelumnya,
- **next:** menunjuk ke kotak sesudahnya.

Bagian 2: Kelas DoubleLinkedList

- **Baris 7–9**

Class ini menyimpan semua kotak (node) dalam bentuk daftar berantai dua arah.

Awalnya, list ini kosong karena head (ujung awal list) masih None.

Fungsi-fungsi dalam DoubleLinkedList

1. Fungsi append(data)

- **Baris 12–21**

Digunakan untuk menambahkan data di bagian akhir list.

- Kalau list masih kosong, data jadi kotak pertama (head).
- Kalau sudah ada isinya, cari sampai ujung lalu tambahkan kotak baru di belakangnya.

2. Fungsi display()

- **Baris 24–28**

Menampilkan isi list dari depan ke belakang.

- Cetak isi setiap kotak diikuti tanda <->.
- Setelah semua, cetak None untuk menandakan akhir list.
-

3. Fungsi delete_from_beginning()

- **Baris 31–37**

Menghapus kotak paling depan (yang pertama).

- Kalau list kosong, tampilkan pesan.
- Kalau ada isinya, pindahkan head ke kotak berikutnya.

4. Fungsi delete_from_end()

- **Baris 40–50**

Menghapus kotak terakhir.

- Kalau kosong, tampilkan pesan.
- Kalau hanya satu kotak, langsung kosongkan list.
- Kalau ada beberapa, cari sampai akhir lalu hapus kotak terakhir.

5. Fungsi delete_by_value(value)

- **Baris 53–69**

Menghapus kotak berdasarkan nilai datanya.

- Telusuri semua kotak satu per satu.
- Kalau ditemukan:
 - Kalau kotak itu yang pertama, pindahkan head.
 - Kalau kotaknya di tengah atau akhir, sambungkan kotak sebelumnya dengan kotak sesudahnya.
- Kalau tidak ketemu, beri pesan bahwa datanya tidak ada.

Contoh Penggunaan Program

- **Baris 72–89**

Program dicoba dengan cara:

- Menambahkan 4 data: 10, 20, 30, 40.
- Menampilkan isi list.
- Menghapus data pertama (10).
- Menghapus data terakhir (40).
- Menghapus data dengan nilai 20.
- Mencoba menghapus data 99 (yang tidak ada), untuk melihat pesan error.