

Nama: Arya Raihan Hanif

Kelas: Praktikum Pemrograman C

NPM: 233040101

Link Repo: https://github.com/aryahanif/PP12025_C_233040101

Elemen List (Node)

Tes 1

Apa fungsi atribut next pada kelas Node?

Jawab: **Fungsi Atribut next pada Kelas Node**

Atribut next pada kelas Node berfungsi untuk menyimpan referensi ke Node berikutnya dalam suatu struktur data yang bersifat linier, seperti **linked list**. Dengan adanya atribut next, kita dapat menghubungkan satu Node dengan Node lainnya, sehingga membentuk suatu rangkaian atau urutan.

Tes 2

Perintah apa yang digunakan untuk menambahkan relasi antar Node?

Jawab: **Perintah untuk Menambahkan Relasi Antar Node**

Untuk menambahkan relasi antar Node, kita menggunakan **setter** dari atribut next, yaitu metode `setNext()`. Berikut adalah contoh cara menambahkan relasi antar Node:

```
//membuat 2 buah node n1 & n2
Node n1 = new Node(5);
Node n2 = new Node(7);
```

```
//membuat relasi Node n1 & n2
n1.setNext(n2);
```

Tes 3

Apa fungsi atribut pointer p yang terdapat di Latihan-2 (NodeMain)?

Jawab: **Fungsi Atribut Pointer p pada NodeMain**

Atribut p pada kelas NodeMain berfungsi sebagai **pointer** atau **iterator** yang digunakan untuk **melintasi (traverse)** rangkaian Node yang telah dibuat. Pointer p membantu kita mengakses setiap Node secara berurutan, mulai dari Node pertama (n1) hingga Node terakhir (yang memiliki next bernilai null).

Tes 4

Ubahlah Latihan-2 (NodeMain) yang telah anda kerjakan sehingga menjadi urutan Node seperti dibawah ini

- a. 5, 7, 9, 8
- b. 2, 3, 5, 7, 9

Jawab: A.

```
package Pertemuan2;
```

```
public class Latihan2A {  
    public static void main(String[] args) {  
  
        //membuat 2 buah node n1 & n2  
        Node n1 = new Node(5);  
        Node n2 = new Node(7);  
        Node n3 = new Node(9);  
        Node n4 = new Node(8);  
  
        //membuat relasi Node n1 & n2  
        n1.setNext(n2);  
        n2.setNext(n3);  
        n3.setNext(n4);  
  
        //menampilkan Node n1 & n2 dengan Pointer p  
        Node p = n1;  
        while(p != null) {  
            System.out.printf("%d ", p.getNilai());  
            p = p.getNext();  
        }  
    }  
}
```

Output:

```
1 package Pertemuan2;
2
3 public class Latihan2A {
4     public static void main(String[] args) {
5
6         //membuat 2 buah node n1 & n2
7         Node n1 = new Node(5);
8         Node n2 = new Node(7);
9         Node n3 = new Node(9);
10        Node n4 = new Node(8);
11
12        //membuat relasi Node n1 & n2
13        n1.setNext(n2);
14        n2.setNext(n3);
15        n3.setNext(n4);
16
17        //menampilkan Node n1 & n2 dengan Pointer p
18        Node p = n1;
19        while(p != null) {
20            System.out.printf("%d ", p.getNilai());
21        }
22    }
23 }
```

5 7 9 8

Jawab: B.

```
package Pertemuan2;
```

```
public class Latihan2B {
    public static void main(String[] args) {
```

```
        //membuat 2 buah node n1 & n2
        Node n1 = new Node(2);
        Node n2 = new Node(3);
        Node n3 = new Node(5);
        Node n4 = new Node(7);
        Node n5 = new Node(9);
```

```
        //membuat relasi Node n1 & n2
        n1.setNext(n2);
        n2.setNext(n3);
        n3.setNext(n4);
        n4.setNext(n5);
```

```
        //menampilkan Node n1 & n2 dengan Pointer p
        Node p = n1;
        while(p != null) {
            System.out.printf("%d ", p.getNilai());
            p = p.getNext();
        }
    }
```

```
}
```

Output:

