

**Laporan Tugas  
Praktikum Pemograman 1**



**Disusun oleh :**

Rudol Prasetyo Sinurat - 233040152

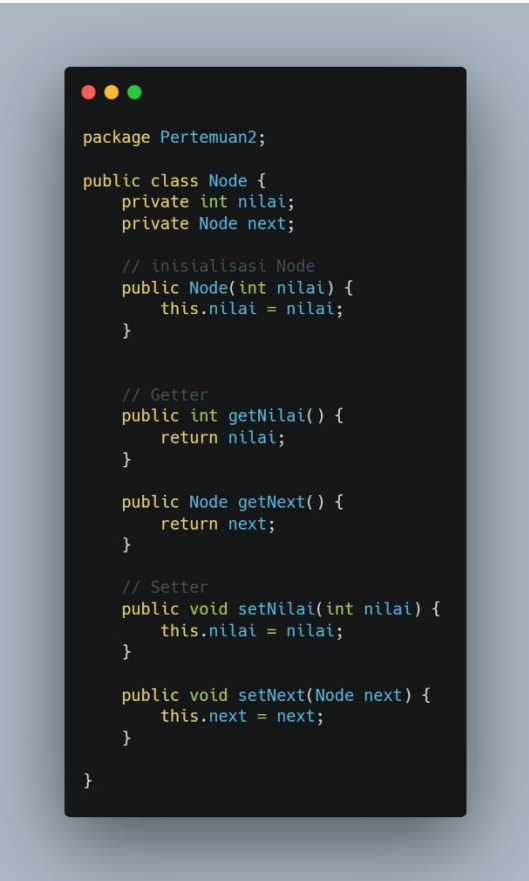
**Dosen Pengampu :**

**ADE SUKENDAR, ST.,MT  
SANDRA ISLAMA PUTRA., S.Si., M.Kom**

**FAKULTAS TEKNIK  
UNIVERSITAS PASUNDAN  
2025**

# Latihan 1

## A. Kode Java Latihan 1



```
package Pertemuan2;

public class Node {
    private int nilai;
    private Node next;

    // inisialisasi Node
    public Node(int nilai) {
        this.nilai = nilai;
    }

    // Getter
    public int getNilai() {
        return nilai;
    }

    public Node getNext() {
        return next;
    }

    // Setter
    public void setNilai(int nilai) {
        this.nilai = nilai;
    }

    public void setNext(Node next) {
        this.next = next;
    }
}
```

## B Penjelasan Mengenai Kode

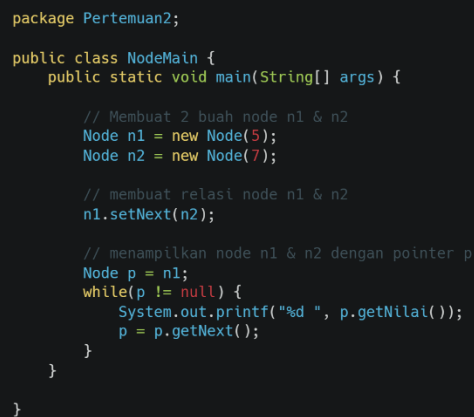
- `private int nilai;` = untuk menyimpan nilai dari node (tipe integer)
- `private Node next;` = untuk menyimpan referensi ke node berikutnya dalam linked list
- `public Node(int nilai) { this.nilai = nilai; }` = untuk menginisialisasi atribut nilai berdasarkan nilai yang diberikan sebagai parameter
- `public void setNilai(int nilai) { this.nilai = nilai; }` = mengubah nilai dari atribut nilai
- `public int getNilai() { return nilai; }` = Mengembalikan nilai yang tersimpan di dalam node
- `public void setNext(Node next) { this.next = next; }` = Menghubungkan node saat ini dengan node lain dalam linked list
- `public Node getNext() { return next; }` = Mengembalikan node berikutnya yang ditunjuk oleh next

## C. Output

A terminal window with a dark background showing the output of a Java program. The output consists of the numbers "5" and "7" separated by a space, displayed in a light blue monospaced font.

## Latihan 2

### A. Kode Java Latihan 2

A screenshot of a code editor window with a dark theme. It contains Java code for a linked list. The code defines a package, a NodeMain class, and a main method. Inside main, it creates two Node objects (n1 with value 5, n2 with value 7), sets n1's next pointer to n2, and then uses a while loop to traverse the list starting from n1, printing the values 5 and 7. The code is as follows:

```
package Pertemuan2;

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

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

        // membuat relasi node n1 & n2
        n1.setNext(n2);

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

### B. Penjelasan

Membuat Node =

- Node n1 = new Node(5);
  - Node n2 = new Node(7);
  - n1.setNext(n2); = menunjuk ke node 2
  - Node p = n1; while (p != null) {
    - System.out.printf("%d", p.getNilai());
    - p = p.getNext();
- } = variable p mulai dari n1, selama p tidak kosong (null), program mencetak nilai dari node yang sedang ditunjuk, kemudian p berpindah ke node berikutnya

## C. Output



```
5 7
```

### Test – 1

\* Untuk menyimpan referensi ke node berikutnya dalam linked list.

### Test – 2

\* `setNext(Node next)`

Contoh penggunaan = `n1.setNext(n2);`

### Test – 3

\* Pointer `p` digunakan untuk menelusuri atau menampilkan nilai-nilai dari node yang telah terhubung dalam struktur linked list

### Test – 4

Kode :

```
package Pertemuan2;

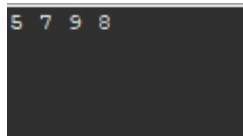
public class NodeMain {
    public static void main(String[] args) {

        // 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 :



```
5 7 9 8
```

Kode :

```
package Pertemuan2;

public class NodeMain {
    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 : 2 3 5 7 9