

Panji Iman Baskoro  
171111023  
Praktikum Progdas 2

## Modul 6

coba6.java

```
public class coba6 {  
    1. import java.util.Scanner;  
    2.  
    3. public class coba6 {  
    4.  
    5.     public static void main(String[] args) {  
    6.         Scanner pot = new Scanner(System.in);  
    7.         System.out.println("Masukkan root nodenya");  
    8.         int kolo = pot.nextInt();  
    9.         Tree t = new Tree(new TreeNode(kolo));  
    10.        System.out.println("berapa childnya?");  
    11.  
    12.        int yolo = pot.nextInt();  
    13.        for(int y = 0; y < yolo; y++){  
    14.            System.out.println("masukkan node Child");  
    15.            int polo = pot.nextInt();  
    16.            System.out.println("masukkan distance Child");  
    17.            int lolo = pot.nextInt();  
    18.            t.root.add_child(new TreeNode(polo), lolo);  
    19.        }  
    20.        t.print();  
    21.        pot.close();  
    22.    }  
    23.  
    24.}
```

Tree .java

```
public class Tree {  
    1.     TreeNode root;  
    2.  
    3.     public Tree() {  
    4.         this.root = null;  
    5.     }  
    6.  
    7.     public Tree(TreeNode root) {  
    8.         this.root = root;  
    9.     }  
    10.  
    11.     void print() {  
    12.         if (this.root == null) {  
    13.             System.out.println();  
    14.         } else {
```

```
15.     this.root.print();
16. }
17. }
18.}
```

## TreeNode.java

```
import java.util.ArrayList;
```

```
1. public class TreeNode {
2.     TreeNode parent;
3.     double distance;
4.     ArrayList<TreeNode> children;
5.     int data;
6.
7.     public TreeNode(int new_data) {
8.         this.data = new_data;
9.         this.parent = null;
10.        this.distance = 0.0;
11.        this.children = new ArrayList<TreeNode>();
12.    }
13.
14.    void set_parent(TreeNode new_parent, double distance) {
15.        this.parent = new_parent;
16.        this.distance = distance;
17.        if (this.parent != null) {
18.            parent.children.add(this);
19.        }
20.    }
21.
22.    void set_parent(TreeNode new_parent) {
23.        this.set_parent(new_parent, 0);
24.    }
25.
26.    void add_child(TreeNode new_child, double distance) {
27.        new_child.set_parent(this);
28.        new_child.distance = distance;
29.    }
30.
31.    /* Simply remove child from this node's children */
32.    void remove_child(TreeNode child) {
33.        this.children.remove(child);
34.    }
35.
36.    void print(String spaces, double distance) {
37.        System.out.println(data+" Distance from Parent "+this.distance+ " distance
from initial node : "+(distance+this.distance));
```

```

38.     for (int i = 0; i < this.children.size(); i++) {
39.         this.children.get(i).print(" ", this.distance);
40.     }
41. }
42.
43. void print() {
44.     this.print("", 0);
45. }
46.}

```

output :

```

budosen@budosen-pc:/mnt/b2c7efbf-ef52-437d-8ca7-e46ea581cbba/Kulia
ertemuan yang tertunda$ java coba6
Masukkan root nodenya
10
berapa childnya?
3
masukkan node Child
2
masukkan distance Child
3
masukkan node Child
4
masukkan distance Child
2
masukkan node Child
3
masukkan distance Child
4
10 Distance from Parent 0.0 distance from initial node : 0.0
2 Distance from Parent 3.0 distance from initial node : 3.0
4 Distance from Parent 2.0 distance from initial node : 2.0
3 Distance from Parent 4.0 distance from initial node : 4.0
budosen@budosen-pc:/mnt/b2c7efbf-ef52-437d-8ca7-e46ea581cbba/Kulia
ertemuan yang tertunda$ 

```

Terimakasih