```java
package binarytree;

// A Java program for in-place conversion of Binary Tree to DLL


class BinaryTree52
{
    Node root;

    // head --> Pointer to head node of created doubly linked list
    Node head;

    // Initialize previously visited node as NULL. This is
    // static so that the same value is accessible in all recursive
    // calls
    static Node prev = null;

    // A simple recursive function to convert a given Binary tree
    // to Doubly Linked List
    // root --> Root of Binary Tree
    void BinaryTree522DoubleLinkedList(Node root)
    {
        // Base case
        if (root == null)
            return;

        // Recursively convert left subtree
        BinaryTree522DoubleLinkedList(root.left);

        // Now convert this node
        if (prev == null)
            head = root;
        else
        {
            root.left = prev;
            prev.right = root;
        }
        prev = root;

        // Finally convert right subtree
        BinaryTree522DoubleLinkedList(root.right);
    }

    /* Function to print nodes in a given doubly linked list */
    void printList(Node node)
    {
        while (node != null)
        {
            System.out.print(node.data + " ");
            node = node.right;
        }
    }

    // Driver program to test above functions
    public static void main(String[] args)
    {
        // Let us create the tree as shown in above diagram
        BinaryTree52 tree = new BinaryTree52();
        tree.root = new Node(10);
        tree.root.left = new Node(12);
        tree.root.right = new Node(15);
        tree.root.left.left = new Node(25);
        tree.root.left.right = new Node(30);
        tree.root.right.left = new Node(36);

        // convert to DLL
```

```
        tree.BinaryTree522DoubleLinkedList(tree.root);

        // Print the converted List
        tree.printList(tree.head);

    }
}
// This code has been contributed by Mayank Jaiswal(mayank_24)
```