**public** **class** BinaryTreeTest {

**public** **static** **void** main(String[] args){

BT bt;

BT lt;

BT rt;

BT leftT;

BT rightT;

lt = **new** BT(**new** BT(), "D", **new** BT());

rt = **new** BT(**new** BT(), "E", **new** BT());

bt = **new** BT(lt, "B", rt);

lt = bt;

rt = **new** BT(**new** BT(), "C", **new** BT());

bt = **new** BT(lt, "A", rt);

System.***out***.println("최상위 노드의 데이터값");

System.***out***.println(bt.rootData());

System.***out***.println("좌측서브트리의 루트노드 데이터값");

leftT = bt.leftTree();

System.***out***.println(leftT.rootData());

System.***out***.println("우즉서브트리의 루트노드 데이터값");

rightT = bt.righrTree();

System.***out***.println(rightT.rootData());

}

}

**class** TreeNode{

Object data;

TreeNode left;

TreeNode right;

}

**class** BT{

TreeNode root;

BT(){

root = **null**;

}

BT(BT left, Object data, BT right){

root = **new** TreeNode();

root.data = data;

root.left = left.root;

root.right = right.root;

}

**boolean** isEmpty() {

**if** (root == **null**)

**return** **true**;

**else**

**return** **false**;

}

BT leftTree(){

BT newTree = **new** BT();

newTree.root = root.left;

**return** newTree;

}

BT righrTree(){

BT newTree = **new** BT();

newTree.root = root.right;

**return** newTree;

}

Object rootData() {

**if** (isEmpty() == **true**)

**return** **null**;

**else**

**return** root.data;

}

}



