DOUBLY LINKED LIST:

**package** doublylinkedlist28;

**public** **class** Doublylinkedlist {

**class** Node{

**int** data;

Node previous;

Node next;

**public** Node(**int** data) {

**this**.data = data;

}

}

Node head, tail = **null**;

**public** **void** addNode(**int** data) {

Node newNode = **new** Node(data);

**if**(head == **null**) {

head = tail = newNode;

head.previous = **null**;

tail.next = **null**;

}

**else** {

tail.next = newNode;

newNode.previous = tail;

tail = newNode;

tail.next = **null**;

}

}

**public** **void** display() {

Node current = head;

**if**(head == **null**) {

System.***out***.println("List is empty");

**return**;

}

System.***out***.println("Nodes of doubly linked list: ");

**while**(current != **null**) {

System.***out***.print(current.data + " ");

current = current.next;

}

}

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

Doublylinkedlist dList = **new** Doublylinkedlist();

dList.addNode(1);

dList.addNode(2);

dList.addNode(3);

dList.addNode(4);

dList.addNode(5);

dList.display();

}

}