T6 (LinkedList) Answer :

**Question 1**

1. Assume that a node class called Node<E> exist. Create two Character nodes called node1 and node2. Node1 contains alphabet ‘a’ and node2 contains alphabet ‘z’.

**Answer:**

Node<Character> node1 = new Node<Character>(‘a’);

Node<Character> node2 = new Node<Character>(‘z’);

1. Draw the nodes from (a).

**Answer:**

node2

node1

a

null

z

null

head

tail

1. Write a statement/code for the head accessing the tail node. Modify 1(b) to show this.

**Answer:**

head.next = tail;

a

next

z

null

head

tail

1. Create a new node, newNode. Add this new node at the first location of all existing nodes. Draw these nodes.

**Answer:**

a

next

z

null

head

tail

head

b

next

1. What are the conditions for this operation?

**Answer:**

1. Only 1 node in list
2. Many nodes

1. Write a list of operations/steps/pseudocode needed to add the newNode to the first location.

**Answer:**

**Condition 1**

1. If tail = null, then tail & head refers the same node

**Condition 2**

1. Create new node object.
2. Assign the new node’s next reference to the current first node(head).
3. Assign the new node as the head.
4. Write codes to assign the newNode to the first location.

**Answer:**

public void addFirst(Character c) {

Node<Character> firstNode = new Node<Character>('b');

firstNode.next = head;

head = firstNode;

size++;

if(tail==null)

tail = head;

}

1. Repeat (d) – (f), for the following operations : (Answers available from lecture notes on LinkedList)
   1. addLast() – value of element, c
   2. add(int index, E e)
   3. removeFirst()
   4. removeLast()
   5. remove(int index, E e)

* If necessary, list the possibilities to be considered for these operations.