# Q2. Dry Run / Trace

You start with an empty ER list.

Perform the following operations step by step, and after each step, draw the doubly linked list.

Show each node’s patientID, and show arrows for both prev and next.

Operations:

1. insertAtEnd(101)

2. insertAtEnd(102)

3. insertAtBeginning(200) // critical patient

4. insertAtPosition(150, 2)

5. deleteFromBeginning()

6. insertAtEnd(300)

After step 6, answer:

(a) What is the patientID at the head?

(b) What is the patientID at the tail?

(c) Write the list from head → tail (forward traversal).

(d) Write the list from tail → head (backward traversal).

Marks are given for correct order and correct pointer understanding.

## 1.insertAtEnd(101)

List: [101]

Head -- > 101 < -- Tail

Head = 101 , Tail = 101

Head -- > prev = NULL, Tail -- > next = NULL

## 2. insertAtEnd(102)

List: [101] -- > < -- [102]

Head -- > prev = NULL , Head -- > next = 102

Head = 101 , Tail = 102

## 3. insertAtBeginning(200)

List: [200] -- > < -- [101] -- > < -- [102]

Head = 200, Tail = 102

Head -- > prev = NULL , Head -- > Next = 101

Tail -- > prev = 101 , Tail -- > next = NULL

## 4. insertAtPosition(150 , 2 )

List: [200] -- > < -- [150] -- > < -- [101] -- > < -- [102]

Head = 200 , Tail = 102

Head -- > prev = NULL , Head -- > Next = 150

Tail -- > prev = 101 , Tail -- > next = NULL

## 5. deleteFromBeginning()

Deleted 200

List: [150] -- > < -- [101] -- > < -- [102]

Head = 150, Tail = 102

Head -- > prev = NULL , Head -- > Next = 101

Tail -- > prev = 101 , Tail -- > next = NULL

## 6. insertAtEnd(300)

List: [150] -- > < -- [101] -- > < -- [102] -- > < -- [300]

Head = 150, Tail = 300

Head -- > prev = NULL , Head -- > Next = 101

Tail -- > prev = 102 , Tail -- > next = NULL

## After step 6:

(a) PatientID at Head: 150

(b) PatientID at Tail: 300

(c) Forward Traversal (Head -- > Tail): 150 -- > 101 -- > 102 -- > 300

(d) Backward Traversal (Tail -- > Head): 300 -- > 102 -- > 101 -- > 150