Double Linked-List console.log output

**Insert Head-List:**

1. *dLinkedList {head: dNode, tail: dNode, size: 7}*
   1. head: dNode
      1. data: 700
      2. next: dNode
         1. data: 600
         2. next: dNode
            1. data: 500
            2. next: dNode

data: 400

next: dNode

data: 300

next: dNode

data: 200

next: dNode

data: 100

next: null

prev: dNode {data: 200, next: dNode, prev: dNode}

\_\_proto\_\_: Object

prev: dNode {data: 300, next: dNode, prev: dNode}

\_\_proto\_\_: Object

prev: dNode {data: 400, next: dNode, prev: dNode}

\_\_proto\_\_: Object

prev: dNode {data: 500, next: dNode, prev: dNode}

\_\_proto\_\_: Object

* + - * 1. prev: dNode {data: 600, next: dNode, prev: dNode}
        2. \_\_proto\_\_: Object
      1. prev: dNode {data: 700, next: dNode, prev: null}
      2. \_\_proto\_\_: Object
    1. prev: null
    2. \_\_proto\_\_: Object
  1. tail: dNode
     1. data: 100
     2. next: null
     3. prev: dNode
        1. data: 200
        2. next: dNode {data: 100, next: null, prev: dNode}
        3. prev: dNode
           1. data: 300
           2. next: dNode {data: 200, next: dNode, prev: dNode}
           3. prev: dNode

data: 400

next: dNode {data: 300, next: dNode, prev: dNode}

prev: dNode

data: 500

next: dNode {data: 400, next: dNode, prev: dNode}

prev: dNode

data: 600

next: dNode {data: 500, next: dNode, prev: dNode}

prev: dNode

data: 700

next: dNode {data: 600, next: dNode, prev: dNode}

prev: null

\_\_proto\_\_: Object

\_\_proto\_\_: Object

\_\_proto\_\_: Object

\_\_proto\_\_: Object

* + - * 1. \_\_proto\_\_: Object
      1. \_\_proto\_\_: Object
    1. \_\_proto\_\_: Object
  1. size: 7
  2. \_\_proto\_\_: Object

Insert Tail List:

1. *dLinkedList {head: dNode, tail: dNode, size: 3}*
   1. head: dNode
      1. data: 100
      2. next: dNode
         1. data: 90
         2. next: dNode
            1. data: 80
            2. next: dNode

data: 70

next: dNode

data: 60

next: dNode

data: 50

next: dNode

data: 40

next: dNode

1. data: 30
2. next: null
3. prev: dNode {data: 40, next: dNode, prev: dNode}
4. \_\_proto\_\_: Object

prev: dNode {data: 50, next: dNode, prev: dNode}

\_\_proto\_\_: Object

prev: dNode {data: 60, next: dNode, prev: dNode}

\_\_proto\_\_: Object

prev: dNode {data: 70, next: dNode, prev: dNode}

\_\_proto\_\_: Object

prev: dNode {data: 80, next: dNode, prev: dNode}

\_\_proto\_\_: Object

* + - * 1. prev: dNode {data: 90, next: dNode, prev: dNode}
        2. \_\_proto\_\_: Object
      1. prev: dNode {data: 100, next: dNode, prev: null}
      2. \_\_proto\_\_: Object
    1. prev: null
    2. \_\_proto\_\_: Object
  1. tail: dNode
     1. data: 30
     2. next: null
     3. prev: dNode
        1. data: 40
        2. next: dNode {data: 30, next: null, prev: dNode}
        3. prev: dNode
           1. data: 50
           2. next: dNode {data: 40, next: dNode, prev: dNode}
           3. prev: dNode

data: 60

next: dNode {data: 50, next: dNode, prev: dNode}

prev: dNode

data: 70

next: dNode {data: 60, next: dNode, prev: dNode}

prev: dNode

data: 80

next: dNode {data: 70, next: dNode, prev: dNode}

prev: dNode

data: 90

next: dNode {data: 80, next: dNode, prev: dNode}

prev: dNode

1. data: 100
2. next: dNode {data: 90, next: dNode, prev: dNode}
3. prev: null
4. \_\_proto\_\_: Object

\_\_proto\_\_: Object

\_\_proto\_\_: Object

\_\_proto\_\_: Object

\_\_proto\_\_: Object

* + - * 1. \_\_proto\_\_: Object
      1. \_\_proto\_\_: Object
    1. \_\_proto\_\_: Object
  1. size: 8
  2. \_\_proto\_\_: Object

insertTail

1. Head List Data
2. 100 1
3. 90 2
4. 80 3
5. 70 4
6. 60 5
7. 50 6
8. 40 7
9. 30 8
10. Tail List Data
11. 30 1
12. 40 2
13. 50 3
14. 60 4
15. 70 5
16. 80 6
17. 90 7
18. 100 8

**insertHead()**

**Head List Data**

700 1

600 2

500 3

400 4

300 5

200 6

100 7

**Tail List Data**

100 1

200 2

300 3

400 4

500 5

600 6

700 7

/////////////////////////////////////////////////////////////////////////////

const dll = new dLinkedList();

dll.insertHead(100);

console.log(dll);

dll.insertHead(200);

console.log(dll);

dll.insertHead(300);

dll.insertHead(400);

dll.insertHead(500);

dll.insertHead(600);

dll.insertHead(700);

console.log(dll);

dll.printHeadListData();

dll.printTailListData();

dll.removeAtHead(3);

console.log(dll);

dll.printHeadListData();

dll.removeAtHead(5);

console.log(dll);

dll.printHeadListData();

dll.removeAtHead(0);

dll.printHeadListData();

dll.removeAtHead(20);

console.log(dll);

dLinkedList {head: dNode, tail: dNode, size: 1, removeAtHead: ƒ}

dLinkedList {head: dNode, tail: dNode, size: 2, removeAtHead: ƒ}

dLinkedList {head: dNode, tail: dNode, size: 7, removeAtHead: ƒ}

Head List Data

700 0

600 1

500 2

400 3

300 4

200 5

100 6

Tail List Data

100 0

200 1

300 2

400 3

500 4

600 5

700 6

Remove at index 3

dLinkedList {head: dNode, tail: dNode, size: 6, removeAtHead: ƒ}

Head List Data

700 0

600 1

500 2

300 3

200 4

100 5

Remove at index 5

dLinkedList {head: dNode, tail: dNode, size: 5, removeAtHead: ƒ}

Head List Data

700 0

600 1

500 2

300 3

200 4

Remove at index 0

Head List Data

600 0

500 1

300 2

200 3

dLinkedList {head: dNode, tail: null, size: 4, removeAtHead: ƒ}

//////////////////////////////////////////////////////////////////////////////////////////////////////////

dll.insertTail(100);

console.log(dll);

dll.insertTail(90);

console.log(dll);

dll.insertTail(80);

dll.insertTail(70);

dll.insertTail(60);

dll.insertTail(50);

dll.insertTail(40);

dll.insertTail(30);

dll.printHeadListData();

dll.printTailListData();

dll.removeAtHead(3);

console.log(dll);

dll.printHeadListData();

dll.removeAtHead(5);

console.log(dll);

dll.printHeadListData();

dll.removeAtHead(0);

dll.printHeadListData();

dll.removeAtHead(20);

console.log(dll);

dll.printTailListData();

dLinkedList {head: dNode, tail: dNode, size: 1, removeAtHead: ƒ}

dLinkedList {head: dNode, tail: dNode, size: 2, removeAtHead: ƒ}

Head List Data

100 0

90 1

80 2

70 3

60 4

50 5

40 6

30 7

Tail List Data

30 0

40 1

50 2

60 3

70 4

80 5

90 6

100 7

Remove Head List at index 3

dLinkedList {head: dNode, tail: dNode, size: 7, removeAtHead: ƒ}

Head List Data

100 0

90 1

linked\_list.js:297 80 2

linked\_list.js:297 60 3

linked\_list.js:297 50 4

40 5

30 6

Remove Head List at index 5

dLinkedList {head: dNode, tail: dNode, size: 6, removeAtHead: ƒ}

Head List Data

100 0

90 1

80 2

60 3

50 4

30 5

Remove Head List at index 0

Head List Data

90 0

80 1

60 2

50 3

30 4

dLinkedList {head: dNode, tail: dNode, size: 5, removeAtHead: ƒ}

Tail List Data

30 0

50 1

60 2

80 3

90 4

////////////////////////////////////////////////////////////////////////////////////

dll.insertTail(100);

console.log(dll);

dll.insertHead(200);

console.log(dll);

dll.insertTail(90);

dll.insertHead(300);

dll.insertTail(80);

dll.insertHead(400);

dll.insertTail(70);

dll.insertHead(500);

dll.printHeadListData();

dll.printTailListData();

dll.removeAtHead(3);

console.log(dll);

dll.printHeadListData();

dll.removeAtHead(6);

console.log(dll);

dll.printHeadListData();

dll.removeAtHead(0);

dll.printHeadListData();

dll.removeAtHead(20);

console.log(dll);

dll.printTailListData();

Insert Tail with data 100

dLinkedList {head: dNode, tail: dNode, size: 1, removeAtHead: ƒ}

Insert Head with data 200

dLinkedList {head: dNode, tail: dNode, size: 2, removeAtHead: ƒ}

Insert Tail with data 90

Insert Head with data 300

Insert Tail with data 80

Insert Head with data 400

Insert Tail with data 70

Insert Head with data 500

Head List Data

500 0

400 1

300 2

200 3

100 4

90 5

80 6

70 7

Tail List Data

70 0

linked\_list.js:289 80 1

linked\_list.js:289 90 2

100 3

200 4

300 5

400 6

500 7

Remove Head List at index 3

dLinkedList {head: dNode, tail: dNode, size: 7, removeAtHead: ƒ}

Head List Data

500 0

400 1

300 2

100 3

90 4

80 5

70 6

Remove Head List at index 6

dLinkedList {head: dNode, tail: dNode, size: 6, removeAtHead: ƒ}

Head List Data

500 0

400 1

300 2

100 3

90 4

80 5

Remove Head List at index 0

Head List Data

400 0

300 1

100 2

90 3

80 4

dLinkedList {head: dNode, tail: dNode, size: 5, removeAtHead: ƒ}

Tail List Data

80 0

90 1

100 2

300 3

400 4

/////////////////////////////////////////////////////////////////////////////////

dll.insertHeadAt(450, 3);

dll.printHeadListData();

dll.insertHeadAt(350, 5);

dll.printHeadListData();

Head List Data

700 0

600 1

500 2

400 3

300 4

200 5

100 6

Insert in Head list data = 450 and index = 3

Head List Data

700 0

600 1

500 2

450 3

400 4

300 5

200 6

100 7

Insert in Head list data = 350 and index = 5

Head List Data

700 0

600 1

500 2

450 3

400 4

350 5

300 6

200 7

100 8