Week 5 Quiz: Linked Lists

Read the code for insert_after, a method of a LinkedList.

```python
def insert_after(self, marker: object, item: object) -> None:
    """Insert <item> after the first time <marker> occurs in this linked list.
    ""
    precondition: <marker> is in this linked list.
    ""
    curr = self._first
    while curr.item != marker:
        curr = curr.next
    insert = _Node(item)
    curr.next = insert
```

Answer the questions below about the following client code.

```python
>>> lst = LinkedList([1, 3, 2, 6])
>>> lst.insert_after(3, 4)
```

1. Unfortunately insert_after has a bug. Draw what lst looks like after the client code is run.

2. Draw what lst should look like after the client code is run.

3. Write a correct implementation of insert_after without using any other linked list methods.