Add Elements at a Specific Index in a Linked List

Let's create a addAt(index,element) method that adds an element at a given index. Just like how we remove elements at a given index, we need to keep track of the currentIndex as we traverse the linked list. When the currentIndex matches the given index, we would need to reassign the previous node's next property to reference the new added node. And the new node should reference the next node in the currentIndex. Returning to the conga line example, a new person wants to join the line, but he wants to join in the middle. You are in the middle of the line, so you take your hands off of the person ahead of you. The new person walks over and puts his hands on the person you once had hands on, and you now have your hands on the new person.

Create an `addAt(index,element)` method that adds an element at a given index. Return false if an element could not be added. **Note:** Remember to check if the given index is a negative or is longer than the length of the linked list.

🧪	Your `addAt` method should reassign `head` to the new node when the given index is 0.
🧪	Your `addAt` method should increase the length of the linked list by one for each new node added to the linked list.
🧪	Your `addAt` method should return `false` if a node was unable to be added.