Reversing a Linked List
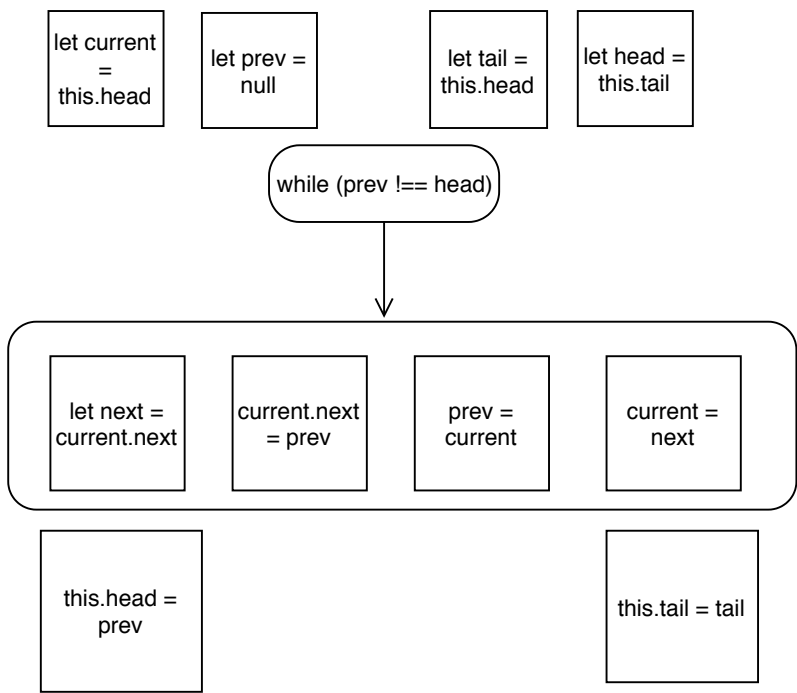
let current = this.head

let prev = null

let tail = this.head

let head = this.tail

while (prev !== head)

let next = current.next

current.next = prev

prev = current

current = next

this.head = prev

this.tail = tail

Example

this.head

```
1 → 2 → 3 → 4 → 5 → 6 → null
```

this.tail

First Loop

next = 2

current.next = null

prev = 1

current = 2

Second Loop

next = 3

current.next = 1

prev = 2

current = 3

Third Loop

next = 4

current.next = 2

prev = 3

current = 4

Fourth Loop

next = 5

current.next = 3

prev = 4

current = 5

Fifth Loop

next = 6

current.next = 4

prev = 5

current = 6

Sixth Loop

next = 7

current.next = 5

prev = 6

current = 7

prev === head, so loop exits

Update reference pointers
this.tail = tail
this.head = prev

this.tail

this.head

```
null ← 1 ← 2 ← 3 ← 4 ← 5 ← 6
```