

```

Run Main
"C:\Program Files\Java\jdk-17\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2\lib\idea_rt.jar=57782:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2\bin"
Lista:
100 -> 200 -> 300 -> 400 -> 600 -> null
Después de cambiar el valor del 3er nodo a 333:
100 -> 200 -> 333 -> 400 -> 600 -> null
Insertar el nodo 700 después del nodo 600:
100 -> 200 -> 333 -> 400 -> 600 -> 700 -> null
Insertar el nodo 50 al principio:
50 -> 100 -> 200 -> 333 -> 400 -> 600 -> 700 -> null

Process finished with exit code 0

```

```

Main.java
1  class Nodo { 12 usages
2      int data; 4 usages
3      Nodo next; 11 usages
4
5      // Constructor para inicializar un nodo
6      public Nodo(int data) { 3 usages
7          this.data = data;
8          this.next = null;
9      }
10
11
12  public class Main {
13      private Nodo head; // Referencia al primer nodo de la lista 10 usages
14
15      // Constructor para inicializar una lista enlazada vacía
16      public Main() { 1 usage
17          this.head = null;
18      }
19
20      // Método para agregar un nodo al final de la lista
21      public void add(int data) {
22          Nodo newNode = new Nodo(data);
23          if (head == null) {
24              head = newNode;
25          } else {
26              Nodo current = head;
27              // Recorrer hasta el último nodo
28              while (current.next != null) {
29                  current = current.next;
30              }
31              // Enlazar el último nodo al nuevo nodo
32              current.next = newNode;

```

17:26 LF UTF-8 4 spaces

```

        current.next = newNode;
    }
}

// Imprimir los elementos de la lista usando while
public void printList() { 4 usages
    Nodo current = head;
    while (current != null) {
        System.out.print(current.data + " -> ");
        current = current.next;
    }
    System.out.println("null");
}

// Método para cambiar el valor de 300 a 333
public void updateThirdNode(int newData) { 1 usage
    if (head == null) return;

    Nodo current = head;
    int count = 1;

    while (current != null && count < 3) {
        current = current.next;
        count++;
    }

    if (current != null) {
        current.data = newData;
    }
}

```

```
62 // Insertar un nodo después de un nodo con un valor específico
```

```
63 public void insertAfter(int afterValue, int newValue) { 1 usage
```

```
64     Nodo current = head;
```

```
65     while (current != null) {
```

```
66         if (current.data == afterValue) {
```

```
67             Nodo newNode = new Nodo(newValue);
```

```
68             newNode.next = current.next;
```

```
69             current.next = newNode;
```

```
70             break;
```

```
71         }
```

```
72         current = current.next;
```

```
73     }
```

```
74 }
```

```
75 // Método para insertar un nodo al principio de la lista
```

```
76 public void insertAtBeginning(int data) { 1 usage
```

```
77     Nodo newNode = new Nodo(data);
```

```
78     newNode.next = head;
```

```
79     head = newNode;
```

```
80 }
```

```
81 public static void main(String[] args) {
```

```
82     Main list = new Main();
```

```
83     // nodos
```

```
84     list.add(100);
```

```
85     list.add(200);
```

```
86     list.add(300);
```

```
92     list.add(300);
93     list.add(400);
94     list.add(600);
95
96     // Imprimir la lista original usando while
97     System.out.println("Lista:");
98     list.printList();
99
100    // Cambiar valor del 3er nodo de 300 a 333
101    list.updateThirdNode( newData: 333);
102    System.out.println("Después de cambiar el valor del 3er nodo de 300 a 333:");
103    list.printList();
104
105    // Insertar 700 después del nodo 600
106    list.insertAfter( afterValue: 600, newValue: 700);
107    System.out.println("Insertar el nodo 700 después del nodo 600:");
108    list.printList();
109
110    // Insertar nodo 50 al principio
111    list.insertAtBeginning( data: 50);
112    System.out.println("Insertar el nodo 50 al principio:");
113    list.printList();
114 }
115 }
116
```