

CLASE LISTA LIGADA

eclipse-workspace - EDD/src/clase2/ListaLigada.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Type Hierarchy

PruebasNodo... Nodojava ListaLigada... SetADT.java EjemploLista... EjemploConj...

```
1 package clase2;
2
3 public class ListaLigada<T> {
4     Nodo<T> head;
5     int tamanio = 0;
6     Nodo<T> currNode;
7
8     public ListaLigada() {
9     }
10
11     public boolean estaVacia(){
12         return head == null;
13     }
14
15     public int getTamanio(){
16         return this.tamanio;
17     }
18
19     public void agregar(T valor){
20         if (this.head == null){
21             this.head = new Nodo<>(valor);
22             this.tamanio++;
23         } else {
24             currNode = this.head;
25             while (currNode.getSiguiente() != null){
26                 currNode = currNode.getSiguiente();
27             }
28             currNode.setSiguiente(new Nodo<>(valor));
29             this.tamanio++;
30         }
31     }
32
33     public int buscar(T valor){
34         currNode = this.head;
35         int contador = 0;
36         while (currNode != null){
37             if(currNode.getData() == valor){
38                 return contador;
39             }
40             contador++;
41             currNode = currNode.getSiguiente();
42         }
43         return -1;
44     }
45 }
```

Task List

Find All Activate...

Outline

- ListaLigada<T>
- head : Nodo<T>
- tamanio : int
- currNode : Nodo<T>
- ListaLigada()
- estaVacia() : boolean
- getTamanio() : int
- agregar(T) : void
- buscar(T) : int
- actualizar(T, T) : boolean
- trasversal() : void
- agregarAlFinal(T) : void
- agregarAlInicio(T) : void
- agregar_despues_de(T, T) : boolean
- eliminar(int) : void
- eliminarElPrimero() : void
- eliminarElFinal() : void
- buscar1(T) : int
- actualizar1(T, T) : boolean

Problems Javadoc Declaration Console

<terminated> EjemploConjuntos [Java Application] C:\Users\lizet\Downloads\eclipse-dsl-2025-06-R-win32-x86_64\eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jr...

45 21 12 3 23 27 52 11

Writable Smart Insert 17 : 6 : 290 290M of 448M

18°C Humo

Buscar

08:05 p. m. 18/09/2025

eclipse-workspace - EDD/src/clase2/ListaLigada.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Type Hierarchy

PruebasNodo... Nodoojava ListaLigada... SetADT.java EjemploLista... EjemploConj...

```
43     return -1;
44 }
45
46 public boolean actualizar(T actual, T nuevo){
47     currNode = this.head;
48     while (currNode != null){
49         if(currNode.getData() == actual){
50             currNode.setData(nuevo);
51             return true;
52         }
53         currNode = currNode.getSiguiente();
54     }
55     return false;
56 }
57 public void trasversal(){
58     currNode = this.head;
59     while (currNode != null){
60         System.out.print(currNode);
61         currNode = currNode.getSiguiente();
62     }
63     System.out.println();
64 }
65
66 public void agregarAlFinal(T valor) {
67     if (this.head == null) {
68         this.head = new Nodo<>(valor);
69         this.tamano++;
70     } else {
71         currNode = this.head;
72         while (currNode.getSiguiente() != null) {
73             currNode = currNode.getSiguiente();
74         }
75         currNode.setSiguiente(new Nodo<>(valor));
76         this.tamano++;
77     }
78 }
79
80 public void agregarAlInicio(T valor) {
81     if (this.head == null) {
82         this.head = new Nodo<>(valor);
83         this.tamano++;
84     } else {
85         Nodo<T> nuevoNodo = new Nodo<>(valor, this.head);

```

Task List

Find All Activate...

Outline

- ListaLigada<T>
 - head : Nodo<T>
 - tamano : int
 - currNode : Nodo<T>
 - ListaLigada()
 - estaVacia() : boolean
 - getTamano() : int
 - agregar(T) : void
 - buscar(T) : int
 - actualizar(T, T) : boolean
 - trasversal() : void
 - agregarAlFinal(T) : void
 - agregarAlInicio(T) : void
 - agregar_despues_de(T, T) : boolean
 - eliminar(int) : void
 - eliminarElPrimero() : void
 - eliminarElFinal() : void
 - buscar1(T) : int
 - actualizar1(T, T) : boolean

Problems @ Javadoc Declaration Console

<terminated> EjemploConjuntos [Java Application] C:\Users\lizet\Downloads\eclipse-dsl-2025-06-R-win32-x86_64\eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jr...

45 21 12 3 23 27 52 11

Writable Smart Insert 17 : 6 : 290 297M of 448M

18°C Humo

Buscar

08:05 p. m. 18/09/2025

eclipse-workspace - EDD/src/clase2/ListaLigada.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Type Hierarchy

PruebasNodo... Nodo.java ListaLigada... SetADT.java EjemploLista... EjemploConj...

```
79
80 public void agregarAlInicio(T valor) {
81     if (this.head == null) {
82         this.head = new Nodo<>(valor);
83         this.tamano++;
84     } else {
85         Nodo<T> nuevoNodo = new Nodo<>(valor, this.head);
86         this.head = nuevoNodo;
87         this.tamano++;
88     }
89 }
90
91 public boolean agregar_despues_de(T valor1, T valor) {
92     if (this.head == null) {
93         return false;
94     }
95     currNode = this.head;
96     while (currNode != null) {
97         if (currNode.getData() == valor1) {
98             Nodo<T> nuevoNodo = new Nodo<>(valor, currNode.getSiguiente());
99             currNode.setSiguiente(nuevoNodo);
100             this.tamano++;
101             return true;
102         }
103         currNode = currNode.getSiguiente();
104     }
105     return false;
106 }
107 public void eliminar(int posicion) {
108     Nodo<T> anterior = null;
109     currNode = this.head;
110     int contador = 0;
111     while (currNode != null) {
112         if (contador == posicion) {
113             if (anterior == null) {
114                 this.head = currNode.getSiguiente();
115             } else {
116                 anterior.setSiguiente(currNode.getSiguiente());
117             }
118             this.tamano--;
119             return;
120         }
121         anterior = currNode;
```

Task List

Find All Activate...

Outline

- ListaLigada<T>
 - head : Nodo<T>
 - tamano : int
 - currNode : Nodo<T>
 - ListaLigada()
 - estaVacia() : boolean
 - getTamano() : int
 - agregar(T) : void
 - buscar(T) : int
 - actualizar(T, T) : boolean
 - trasversal() : void
 - agregarAlFinal(T) : void
 - agregarAlInicio(T) : void
 - agregar_despues_de(T, T) : boolean
 - eliminar(int) : void
 - eliminarElPrimero() : void
 - eliminarElFinal() : void
 - buscar1(T) : int
 - actualizar1(T, T) : boolean

Problems Javadoc Declaration Console

<terminated> EjemploConjuntos [Java Application] C:\Users\lizet\Downloads\eclipse-dsl-2025-06-R-win32-x86_64\eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre...

45 21 12 3 23 27 52 11

Writable Smart Insert 17 : 6 : 290 305M of 448M

18°C Humo 08:05 p.m. 18/09/2025

eclipse-workspace - EDD/src/clase2/ListaLigada.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Type Hierarchy

PruebasNodo... Nodo.java ListaLigada... SetADT.java EjemploLista... EjemploConj...

```
121 anterior = currNode;
122 currNode = currNode.getSiguiente();
123 contador++;
124 }
125 }
126 public void eliminarElPrimero() {
127     if (this.head != null) {
128         this.head = this.head.getSiguiente();
129         this.tamano--;
130     }
131 }
132 }
133 public void eliminarElFinal() {
134     if (this.head == null) {
135         return;
136     }
137     if (this.head.getSiguiente() == null) {
138         this.head = null;
139         this.tamano--;
140         return;
141     }
142     currNode = this.head;
143     while (currNode.getSiguiente().getSiguiente() != null) {
144         currNode = currNode.getSiguiente();
145     }
146     currNode.setSiguiente(null);
147     this.tamano--;
148 }
149 }
150 public int buscar1(T valor){
151     currNode = this.head;
152     int contador = 0;
153     while (currNode != null){
154         if(currNode.getData() == valor){
155             return contador;
156         }
157         contador++;
158         currNode = currNode.getSiguiente();
159     }
160     return -1;
161 }
162 }
163 public boolean actualizar1(T actual, T nuevo){
```

Task List

Find All Activate...

Outline

- ListaLigada<T>
 - head : Nodo<T>
 - tamano : int
 - currNode : Nodo<T>
 - ListaLigada()
 - estaVacia() : boolean
 - getTamano() : int
 - agregar(T) : void
 - buscar(T) : int
 - actualizar(T, T) : boolean
 - trasversal() : void
 - agregarAlFinal(T) : void
 - agregarAlInicio(T) : void
 - agregar_despues_de(T, T) : boolean
 - eliminar(int) : void
 - eliminarElPrimero() : void
 - eliminarElFinal() : void
 - buscar1(T) : int
 - actualizar1(T, T) : boolean

Problems Javadoc Declaration Console

<terminated> EjemploConjuntos [Java Application] C:\Users\lizet\Downloads\eclipse-dsl-2025-06-R-win32-x86_64\eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre...
45 21 12 3 23 27 52 11

Writable Smart Insert 17 : 6 : 290 313M of 448M

18°C Humo 08:05 p.m. 18/09/2025

eclipse-workspace - EDD/src/clase2/ListaLigada.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Type Hierarchy

PruebasNodo... Nodo.java ListaLigada... SetADT.java EjemploLista... EjemploConj...

```
141 }
142 currNode = this.head;
143 while (currNode.getSiguiete().getSiguiete() != null) {
144     currNode = currNode.getSiguiete();
145 }
146 currNode.setSiguiete(null);
147 this.tamano--;
148 }
149
150 public int buscar1(T valor){
151     currNode = this.head;
152     int contador = 0;
153     while (currNode != null){
154         if(currNode.getData() == valor){
155             return contador;
156         }
157         contador++;
158         currNode = currNode.getSiguiete();
159     }
160     return -1;
161 }
162
163 public boolean actualizar1(T actual, T nuevo){
164     currNode = this.head;
165     while (currNode != null){
166         if(currNode.getData() == actual){
167             currNode.setData(nuevo);
168             return true;
169         }
170         currNode = currNode.getSiguiete();
171     }
172     return false;
173 }
174 public void transversal1(){
175     currNode = this.head;
176     while (currNode != null){
177         System.out.print(currNode);
178         currNode = currNode.getSiguiete();
179     }
180     System.out.println();
181 }
182
183 }
```

Task List

Find All Activate...

Outline

- ListaLigada<T>
 - head : Nodo<T>
 - tamano : int
 - currNode : Nodo<T>
 - ListaLigada()
 - estaVacia() : boolean
 - getTamano() : int
 - agregar(T) : void
 - buscar(T) : int
 - actualizar(T, T) : boolean
 - transversal() : void
 - agregarAlFinal(T) : void
 - agregarAlInicio(T) : void
 - agregar_despues_de(T, T) : boolean
 - eliminar(int) : void
 - eliminarElPrimero() : void
 - eliminarElFinal() : void
 - buscar1(T) : int
 - actualizar1(T, T) : boolean

Problems @ Javadoc Declaration Console

<terminated> EjemploConjuntos [Java Application] C:\Users\lizet\Downloads\eclipse-dsl-2025-06-R-win32-x86_64\eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jv...

45 21 12 3 23 27 52 11

Writable Smart Insert 17 : 6 : 290 317M of 448M

18°C Humo

Buscar

08:06 p. m. 18/09/2025

CLASE PRUEBA

eclipse-workspace - EDD/src/clase2/EjemploListaLigada.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Type Hierarchy

java.security.sasl - C:\Users\lizet\Downloads\ec...
java.smartcardio - C:\Users\lizet\Downloads\ec...
java.sql - C:\Users\lizet\Downloads\eclipse-dsl...
java.sql.rowset - C:\Users\lizet\Downloads\ecli...
java.transaction.xa - C:\Users\lizet\Downloads\...
java.xml - C:\Users\lizet\Downloads\eclipse-d...
java.xml.crypto - C:\Users\lizet\Downloads\ec...
jdk.accessibility - C:\Users\lizet\Downloads\ec...
jdk.attach - C:\Users\lizet\Downloads\eclipse-...
jdk.charsets - C:\Users\lizet\Downloads\ecli...
jdk.compiler - C:\Users\lizet\Downloads\ecli...
jdk.crypto.cryptoki - C:\Users\lizet\Downloads\...
jdk.crypto.ec - C:\Users\lizet\Downloads\ecli...
jdk.crypto.mscapi - C:\Users\lizet\Downloads\...
jdk.dynalink - C:\Users\lizet\Downloads\ecli...
jdk.editpad - C:\Users\lizet\Downloads\eclipse...
jdk.hotspot.agent - C:\Users\lizet\Downloads\...
jdk.httpserver - C:\Users\lizet\Downloads\ecli...
jdk.internal.ed - C:\Users\lizet\Downloads\ecli...
jdk.internal.jvstat - C:\Users\lizet\Downloads...
jdk.internal.le - C:\Users\lizet\Downloads\ecli...
jdk.internal.opt - C:\Users\lizet\Downloads\ec...
jdk.internal.vm.ci - C:\Users\lizet\Downloads\...
jdk.internal.vm.compiler - C:\Users\lizet\Downl...
jdk.internal.vm.compiler.management - C:\Use...
jdk.jartool - C:\Users\lizet\Downloads\ecli...
jdk.javadoc - C:\Users\lizet\Downloads\ecli...
jdk.jcmd - C:\Users\lizet\Downloads\ecli...
jdk.jconsole - C:\Users\lizet\Downloads\ecli...
jdk.jdeps - C:\Users\lizet\Downloads\ecli...
jdk.jdi - C:\Users\lizet\Downloads\ecli...
jdk.jdp.agent - C:\Users\lizet\Downloads\ecli...
jdk.jfr - C:\Users\lizet\Downloads\ecli...
jdk.jshell - C:\Users\lizet\Downloads\ecli...
jdk.jsobject - C:\Users\lizet\Downloads\ecli...
jdk.jstatd - C:\Users\lizet\Downloads\ecli...
jdk.localedata - C:\Users\lizet\Downloads\ecli...
jdk.management - C:\Users\lizet\Downloads\ec...

```
4 public static void main(String[] args) {  
5     ListaLigada<Integer> lista = new ListaLigada<>();  
6     System.out.println("La lista esta vacia? " + lista.estaVacia());  
7     lista.agregar(20);  
8     lista.agregar(17);  
9     lista.agregar(36);  
10    lista.agregar(15);  
11  
12    System.out.println("\nLa lista esta vacia? " + lista.estaVacia());  
13  
14    System.out.println("\nTamaño de la lista: " + lista.getTamaño());  
15  
16    System.out.println("\nContenido: ");  
17    lista.trasversal();  
18  
19    System.out.println("\nAgregar 99 al inicio");  
20    lista.agregarAlInicio(99);  
21  
22    System.out.println("Agregar 33 al final");  
23    lista.agregarAlFinal(33);  
24  
25    System.out.println("Agregar 66 después de 17: ");  
26    lista.agregar_despues_de(17, 66);  
27    lista.trasversal();  
28  
29    System.out.println("\nSustituir 22 en 36");  
30    lista.actualizar(36, 22);  
31    lista.trasversal();  
32  
33    System.out.println("\nTamaño: " + lista.tamaño);  
34    lista.eliminar(2);  
35  
36    System.out.println("\n Eliminar 17: ");  
37    lista.trasversal();  
38  
39    System.out.println("\nEliminar primer y último dato: ");  
40    lista.eliminarElFinal();  
41    lista.eliminarElPrimero();  
42    lista.trasversal();  
43  
44    System.out.println("\nTamaño Final: " + lista.tamaño);  
45  
46 }
```

Task List

Find

Outline

clase2
EjemploListaLigada
main(String[]) : void

Problems Javadoc Declaration Console

<terminated> EjemploConjuntos [Java Application] C:\Users\lizet\Downloads\ecclipse-dsl-2025-06-R-win32-x86_64\ecclipse\plugins\org.eclipse.justi.openjdk.hotspot.j...
45 21 12 3 23 27 52 11

Writable Smart Insert 43 : 9 : 1316 387M of 448M

18°C Humo 08:07 p. m. 18/09/2025

CÓDIGO COMPILADO

The screenshot displays the Eclipse IDE interface with the following components:

- Package Explorer:** Shows a project named 'clase2' containing a file 'EjemploListaLigada.java'.
- Editor:** Displays the source code of 'EjemploListaLigada.java'. The code is as follows:

```
4 public static void main(String[] args) {  
5     ListaLigada<Integer> lista = new ListaLigada<>();  
6     System.out.println("La lista esta vacia? " + lista.estaVacia());  
7     lista.agregar(20);  
8     lista.agregar(17);  
9     lista.agregar(36);  
10    lista.agregar(15);  
11  
12    System.out.println("\nLa lista esta vacia? " + lista.estaVacia());  
13  
14    System.out.println("\nTamaño de la lista: " + lista.getTamaño());  
15 }
```
- Console:** Shows the output of the program execution:

```
<terminated> EjemploListaLigada [Java Application] C:\Users\lizet\Downloads\eclipse-dsl-2025-06-R-win32-x86_64\eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64.jre\bin\java.exe  
La lista esta vacia? true  
La lista esta vacia? false  
Tamaño de la lista: 4  
Contenido:  
|20|->|17|->|36|->|15|->  
Agregar 99 al inicio  
Agregar 33 al final  
Agregar 66 después de 17:  
|99|->|20|->|17|->|66|->|36|->|15|->|33|->  
Sustituir 22 en 36  
|99|->|20|->|17|->|66|->|22|->|15|->|33|->  
Tamaño: 7  
Eliminar 17:  
|99|->|20|->|66|->|22|->|15|->|33|->  
Eliminar primer y último dato:  
|20|->|66|->|22|->|15|->  
Tamaño Final: 4
```
- Outline:** Shows the class structure with 'EjemploListaLigada' and its 'main' method.
- Task List:** Empty.
- Bottom Bar:** Shows 'Writable', 'Smart Insert', and memory usage '43 : 9 : 1316' and '286M of 448M'.