

# LISTA LIGADA

## Tarea 4



# MAIN

```
SmartPhone.java  Nodo.java  Main.java x
1  public class Main {
2      public static void main(String[] args) {
3          Listaligada lista = new Listaligada();
4
5          SmartPhone sp1 = new SmartPhone( marca: "Apple", modelo: "iPhone 13", precio: 16999);
6          SmartPhone sp2 = new SmartPhone( marca: "Samsung", modelo: "Galaxy S21", precio: 13599);
7          SmartPhone sp3 = new SmartPhone( marca: "Google", modelo: "Pixel 6", precio: 11899);
8          SmartPhone sp4 = new SmartPhone( marca: "OnePlus", modelo: "9 Pro", precio: 16489);
9          SmartPhone sp5 = new SmartPhone( marca: "Sony", modelo: "Xperia 1 III", precio: 22099);
10
11         lista.agregarAlFinal(sp1);
12         lista.agregarAlFinal(sp2);
13         lista.agregarAlFinal(sp3);
14         lista.agregarAlFinal(sp4);
15         lista.agregarAlFinal(sp5);
16
17         System.out.println("Contenido de la lista:");
18         lista.transversal();
19
20         lista.eliminar( posicion: 2);
21         System.out.println("\nContenido de la lista después de eliminar la posición 2:");
22         lista.transversal();
23
24         SmartPhone nuevoSP = new SmartPhone( marca: "Huawei", modelo: "P50 Pro", precio: 20399);
25         lista.actualizar(sp2, nuevoSP);
26         System.out.println("\nContenido de la lista después de actualizar el segundo elemento:");
27         lista.transversal();
28
29     }
30 }

taLigada > src > Main > main 30:69 LF UTF-8 4 spaces
```

```
SmartPhone.java  Nodo.java  Main.java x
1  public class Main {
2      public static void main(String[] args) {
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17         System.out.println("Contenido de la lista:");
18         lista.transversal();
19
20         lista.eliminar( posicion: 2);
21         System.out.println("\nContenido de la lista después de eliminar la posición 2:");
22         lista.transversal();
23
24         SmartPhone nuevoSP = new SmartPhone( marca: "Huawei", modelo: "P50 Pro", precio: 20399);
25         lista.actualizar(sp2, nuevoSP);
26         System.out.println("\nContenido de la lista después de actualizar el segundo elemento:");
27         lista.transversal();
28
29         SmartPhone sp6 = new SmartPhone( marca: "Xiaomi", modelo: "Mi 11", precio: 12749);
30         SmartPhone sp7 = new SmartPhone( marca: "Oppo", modelo: "Find X3 Pro", precio: 19549);
31         lista.agregarAlInicio(sp6);
32         lista.agregarAlFinal(sp7);
33         System.out.println("\nContenido de la lista después de agregar un elemento al inicio y otro al final:");
34         lista.transversal();
35
36         lista.eliminarElPrimero();
37         System.out.println("\nContenido de la lista después de eliminar el primer elemento:");
38         lista.transversal();
39     }
40 }
41

taLigada > src > Main > main 30:69 LF UTF-8 4 spaces
```

# CLASS NODO

```
SmartPhone.java  Nodo.java  Main.java
1 public class Nodo { 15 usages
2     SmartPhone dato; 6 usages
3     Nodo siguiente; 22 usages
4
5     public Nodo(SmartPhone dato) { 3 usages
6         this.dato = dato;
7         this.siguiente = null;
8     }
9 }
10
11 class Listaligada { 2 usages
12     private Nodo head; 17 usages
13     private int tamaño; 10 usages
14
15     public Listaligada() { 1 usage
16         head = null;
17         tamaño = 0;
18     }
19
20     public boolean estaVacía() { 4 usages
21         return head == null;
22     }
23
24     public int getTamaño() { no usages
25         return tamaño;
26     }
27
28     public void agregarAlFinal(SmartPhone valor) { 6 usages
```

The screenshot shows an IDE with three tabs: SmartPhone.java, Nodo.java, and Main.java. The Nodo.java tab is active, displaying the following Java code:

```
11 class Listaligada { 2 usages
12
13     public void agregarAlFinal(SmartPhone valor) { 6 usages
14         Nodo nuevo = new Nodo(valor);
15         if (estaVacia()) {
16             head = nuevo;
17         } else {
18             Nodo aux = head;
19             while (aux.siguiete != null) {
20                 aux = aux.siguiete;
21             }
22             aux.siguiete = nuevo;
23         }
24         tamano++;
25     }
26
27     public void agregarAlInicio(SmartPhone valor) { 1 usage
28         Nodo nuevo = new Nodo(valor);
29         if (!estaVacia()) {
30             nuevo.siguiete = head;
31         }
32         head = nuevo;
33         tamano++;
34     }
35
36     public void agregarDespuesDe(SmartPhone referencia, SmartPhone valor) { no usages
37         Nodo aux = head;
38         while (aux != null && !aux.dato.equals(referencia)) {
39             aux = aux.siguiete;
40         }
41         aux.siguiete = new Nodo(valor);
42     }
43 }
```

The bottom status bar shows the file path: Listaligada > src > Nodo.java, and the encoding/indentation settings: 10:1 CRLF UTF-8 4 spaces.

# CLASS NODO

```
SmartPhone.java  Nodo.java x  Main.java
11  class Listaligada { 2 usages
51  public void agregarDespuesDe(SmartPhone referencia, SmartPhone valor) { no usages
52      Nodo aux = head;
53      while (aux != null && !aux.dato.equals(referencia)) {
54          aux = aux.siguiente;
55      }
56      if (aux != null) {
57          Nodo nuevo = new Nodo(valor);
58          nuevo.siguiente = aux.siguiente;
59          aux.siguiente = nuevo;
60          tamaño++;
61      }
62  }
63
64  public void eliminar(int posicion) { 1 usage
65      if (posicion >= 0 && posicion < tamaño) {
66          if (posicion == 0) {
67              head = head.siguiente;
68          } else {
69              Nodo aux = head;
70              for (int i = 0; i < posicion - 1; i++) {
71                  aux = aux.siguiente;
72              }
73              aux.siguiente = aux.siguiente.siguiente;
74          }
75          tamaño--;
76      }
77  }
78  }
79  }
80  }
81  }
82  }
83  }
84  }
85  }
86  }
87  }
88  }
89  }
90  }
91  }
92  }
93  }
94  }
95  }
96  }
97  }
98  }
99  }
100 }
101 }
102 }
103 }
104 }
105 }
106 }
107 }
108 }
109 }
110 }
111 }
112 }
113 }
114 }
115 }
116 }
117 }
118 }
119 }
120 }
121 }
122 }
123 }
124 }
125 }
126 }
127 }
128 }
129 }
130 }
131 }
132 }
133 }
134 }
135 }
136 }
137 }
138 }
139 }
140 }
141 }
142 }
143 }
144 }
145 }
146 }
147 }
148 }
149 }
150 }
151 }
152 }
153 }
154 }
155 }
156 }
157 }
158 }
159 }
160 }
161 }
162 }
163 }
164 }
165 }
166 }
167 }
168 }
169 }
170 }
171 }
172 }
173 }
174 }
175 }
176 }
177 }
178 }
179 }
180 }
181 }
182 }
183 }
184 }
185 }
186 }
187 }
188 }
189 }
190 }
191 }
192 }
193 }
194 }
195 }
196 }
197 }
198 }
199 }
200 }
```

```
SmartPhone.java  Nodo.java x  Main.java
11  class Listaligada { 2 usages
78  }
79  }
80  }
81  }
82  }
83  }
84  }
85  }
86  public void eliminarElFinal() { no usages
87      if (!estaVacia()) {
88          if (tamaño == 1) {
89              head = null;
90          } else {
91              Nodo aux = head;
92              while (aux.siguiente.siguiente != null) {
93                  aux = aux.siguiente;
94              }
95              aux.siguiente = null;
96          }
97          tamaño--;
98      }
99  }
100 }
101 public int buscar(SmartPhone valor) { no usages
102     Nodo aux = head;
103     int posicion = 0;
104     while (aux != null) {
105         if (aux.dato.equals(valor)) {
106             return posicion;
107         }
108         aux = aux.siguiente;
109         posicion++;
110     }
111     return -1;
112 }
113 }
114 }
115 }
116 }
117 }
118 }
119 }
120 }
121 }
122 }
123 }
124 }
125 }
126 }
127 }
128 }
129 }
130 }
131 }
132 }
133 }
134 }
135 }
136 }
137 }
138 }
139 }
140 }
141 }
142 }
143 }
144 }
145 }
146 }
147 }
148 }
149 }
150 }
151 }
152 }
153 }
154 }
155 }
156 }
157 }
158 }
159 }
160 }
161 }
162 }
163 }
164 }
165 }
166 }
167 }
168 }
169 }
170 }
171 }
172 }
173 }
174 }
175 }
176 }
177 }
178 }
179 }
180 }
181 }
182 }
183 }
184 }
185 }
186 }
187 }
188 }
189 }
190 }
191 }
192 }
193 }
194 }
195 }
196 }
197 }
198 }
199 }
200 }
```



# CLASS NODO

```
SmartPhone.java  Nodo.java x  Main.java
11  class Listaligada { 2 usages
100
101  public int buscar(SmartPhone valor) { no usages
102      Nodo aux = head;
103      int posicion = 0;
104      while (aux != null) {
105          if (aux.dato.equals(valor)) {
106              return posicion;
107          }
108          aux = aux.siguiente;
109          posicion++;
110      }
111      return -1;
112  }
113
114  public void actualizar(SmartPhone aBuscar, SmartPhone valor) { 1 usage
115      Nodo aux = head;
116      while (aux != null) {
117          if (aux.dato.equals(aBuscar)) {
118              aux.dato = valor;
119              return;
120          }
121          aux = aux.siguiente;
122      }
123  }
124
125  public void transversal() { 5 usages
126      Nodo aux = head;
```

taLigada > src > © Nodo.java > © Listaligada > eliminarElFinal 86:17 CRLF UTF-8 4 spaces

```
SmartPhone.java  Nodo.java x  Main.java
11  class Listaligada { 2 usages
112  }
113
114  public void actualizar(SmartPhone aBuscar, SmartPhone valor) { 1 usage
115      Nodo aux = head;
116      while (aux != null) {
117          if (aux.dato.equals(aBuscar)) {
118              aux.dato = valor;
119              return;
120          }
121          aux = aux.siguiente;
122      }
123  }
124
125  public void transversal() { 5 usages
126      Nodo aux = head;
127      while (aux != null) {
128          System.out.println(aux.dato);
129          aux = aux.siguiente;
130      }
131  }
132  }
133
```

taLigada > src > © Nodo.java > © Listaligada > eliminarElFinal 86:17 CRLF UTF-8 4 spaces

# CLASS SMART PHONE

© SmartPhone.java × © Nodo.java © Main.java

1 public class SmartPhone { 25 usages

2 private String marca; 2 usages

3 private String modelo; 2 usages

4 private int precio; 2 usages

5

6 public SmartPhone(String marca, String modelo, int precio) { 8 usages

7 this.marca = marca;

8 this.modelo = modelo;

9 this.precio = precio;

10 }

11

12 @Override

13 public String toString() {

14 return "SmartPhone: " + marca + '\'' + " --- modelo: '" + modelo + '\'' + " --- precio= " + precio;

15 }

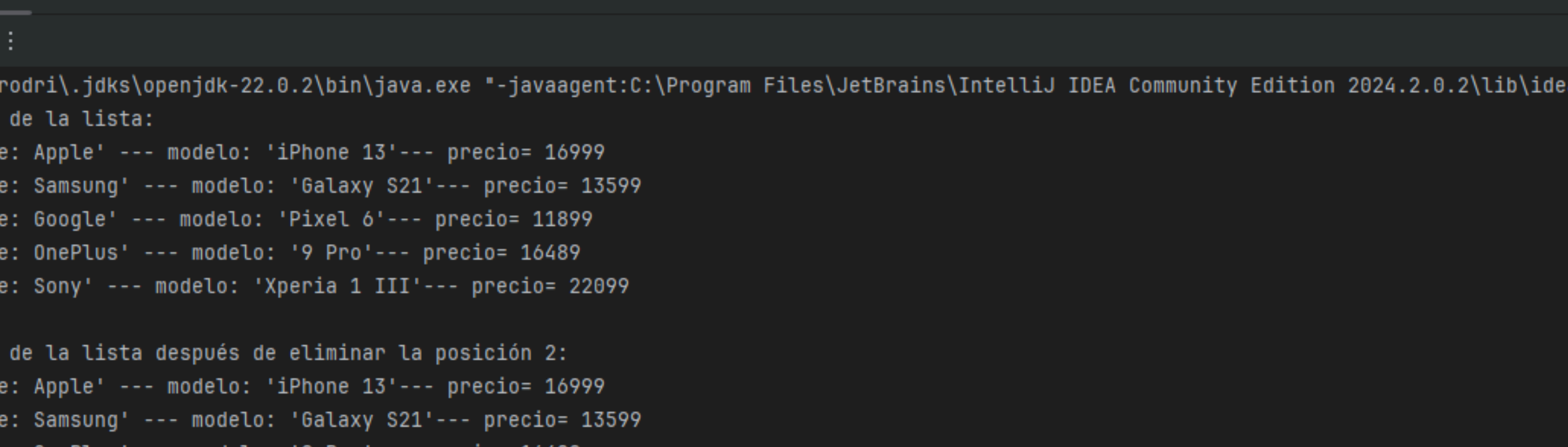
16

17

18 }

⚠ 3 ✓ 6 ^ v

# PANTALLA DE EJECUCION



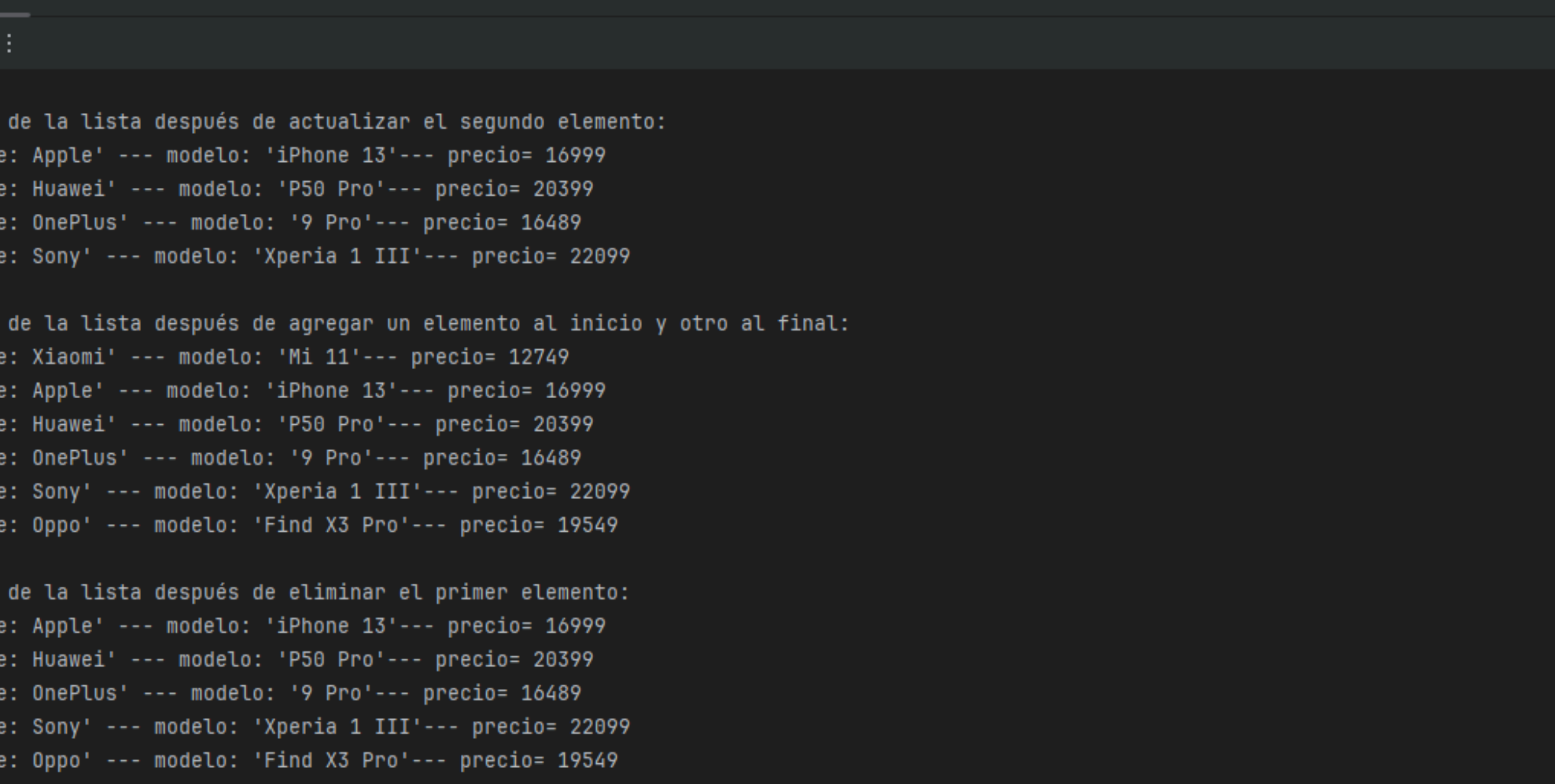
The screenshot shows an IDE with three tabs: SmartPhone.java, Nodo.java, and Main.java. The Run tab is active, displaying the output of a Java program. The output shows the initial list of SmartPhone objects, followed by the list after removing the second element, then after updating the second element, and finally after adding elements to the beginning and end.

```
C:\Users\rodri\.jdk\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.0.2\lib\idea_rt.jar=60763:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.0.2\bin" -Dfile.encoding=UTF-8
Contenido de la lista:
SmartPhone: Apple' --- modelo: 'iPhone 13'--- precio= 16999
SmartPhone: Samsung' --- modelo: 'Galaxy S21'--- precio= 13599
SmartPhone: Google' --- modelo: 'Pixel 6'--- precio= 11899
SmartPhone: OnePlus' --- modelo: '9 Pro'--- precio= 16489
SmartPhone: Sony' --- modelo: 'Xperia 1 III'--- precio= 22099

Contenido de la lista después de eliminar la posición 2:
SmartPhone: Apple' --- modelo: 'iPhone 13'--- precio= 16999
SmartPhone: Samsung' --- modelo: 'Galaxy S21'--- precio= 13599
SmartPhone: OnePlus' --- modelo: '9 Pro'--- precio= 16489
SmartPhone: Sony' --- modelo: 'Xperia 1 III'--- precio= 22099

Contenido de la lista después de actualizar el segundo elemento:
SmartPhone: Apple' --- modelo: 'iPhone 13'--- precio= 16999
SmartPhone: Huawei' --- modelo: 'P50 Pro'--- precio= 20399
SmartPhone: OnePlus' --- modelo: '9 Pro'--- precio= 16489
SmartPhone: Sony' --- modelo: 'Xperia 1 III'--- precio= 22099

Contenido de la lista después de agregar un elemento al inicio y otro al final:
SmartPhone: Xiaomi' --- modelo: 'Mi 11'--- precio= 12749
SmartPhone: Apple' --- modelo: 'iPhone 13'--- precio= 16999
SmartPhone: Huawei' --- modelo: 'P50 Pro'--- precio= 20399
SmartPhone: OnePlus' --- modelo: '9 Pro'--- precio= 16489
```



```
SmartPhone.java  Nodo.java  Main.java x
Run  Main x
Contentido de la lista después de actualizar el segundo elemento:
SmartPhone: Apple' --- modelo: 'iPhone 13'--- precio= 16999
SmartPhone: Huawei' --- modelo: 'P50 Pro'--- precio= 20399
SmartPhone: OnePlus' --- modelo: '9 Pro'--- precio= 16489
SmartPhone: Sony' --- modelo: 'Xperia 1 III'--- precio= 22099

Contentido de la lista después de agregar un elemento al inicio y otro al final:
SmartPhone: Xiaomi' --- modelo: 'Mi 11'--- precio= 12749
SmartPhone: Apple' --- modelo: 'iPhone 13'--- precio= 16999
SmartPhone: Huawei' --- modelo: 'P50 Pro'--- precio= 20399
SmartPhone: OnePlus' --- modelo: '9 Pro'--- precio= 16489
SmartPhone: Sony' --- modelo: 'Xperia 1 III'--- precio= 22099
SmartPhone: Oppo' --- modelo: 'Find X3 Pro'--- precio= 19549

Contentido de la lista después de eliminar el primer elemento:
SmartPhone: Apple' --- modelo: 'iPhone 13'--- precio= 16999
SmartPhone: Huawei' --- modelo: 'P50 Pro'--- precio= 20399
SmartPhone: OnePlus' --- modelo: '9 Pro'--- precio= 16489
SmartPhone: Sony' --- modelo: 'Xperia 1 III'--- precio= 22099
SmartPhone: Oppo' --- modelo: 'Find X3 Pro'--- precio= 19549

Process finished with exit code 0
```