

[Home](#)[About](#)[Content](#)[Others](#)

# EJERCICIOS CON NODOARBOL<T>

[Home](#)[About](#)[Content](#)[Others](#)

The screenshot shows a code editor interface with a dark theme. At the top, there are tabs for "Main.java" and "NodoArbol.java". The "NodoArbol.java" tab is currently selected, indicated by a blue underline. The code itself is a Java class definition for a binary tree node ("NodoArbol"). It includes imports, a constructor that takes data and two children, and methods to get and set the data.

```
1 import java.util.Objects;
2
3 public class NodoArbol { 30 usages new *
4     private Object dato; 8 usages
5     private NodoArbol hijoIzquierdo; 4 usages
6     private NodoArbol hijoDerecho; 4 usages
7
8     public NodoArbol() { no usages new *
9     }
10
11    public NodoArbol(Object dato) { 5 usages new *
12        this.dato = dato;
13    }
14
15    public NodoArbol(Object dato, NodoArbol hijoIzquierdo, NodoArbol hijoDerecho) { 5 usages new *
16        this.dato = dato;
17        this.hijoIzquierdo = hijoIzquierdo;
18        this.hijoDerecho = hijoDerecho;
19    }
20
21    public Object getData() { no usages new *
22        return dato;
23    }
24
25    public void setData(Object dato) { no usages new *
26        this.dato = dato;
27    }
28
```

At the bottom of the code editor, there is a status bar showing file paths ("NodoArbol > src > © NodoArbol"), encoding ("2:1 CRLF"), character set ("UTF-8"), and indentation ("4 spaces"). There is also a small icon of a document with a question mark.

[Home](#)[About](#)[Content](#)[Others](#)

```
>Main.java  © NodoArbol.java ×

3  public class NodoArbol { 30 usages new *
29
29      public NodoArbol getHijoIzquierdo() { no usages new *
30          return hijoIzquierdo;
31      }
32
33      public void setHijoIzquierdo(NodoArbol hijoIzquierdo) { no usages new *
34          this.hijoIzquierdo = hijoIzquierdo;
35      }
36
37      public NodoArbol getHijoDerecho() { no usages new *
38          return hijoDerecho;
39      }
40
41      public void setHijoDerecho(NodoArbol hijoDerecho) { no usages new *
42          this.hijoDerecho = hijoDerecho;
43      }
44
45      @Override new *
46  ©↑      public boolean equals(Object object) {
47          if (this == object) return true;
48          if (object == null || getClass() != object.getClass()) return false;
49          NodoArbol nodoArbol = (NodoArbol) object;
50          return Objects.equals(data, nodoArbol.data);
51      }
52
53      @Override new *
54  ©↑      public int hashCode() {
```

[Home](#)[About](#)[Content](#)[Others](#)

```
>Main.java  © NodoArbol.java ×

3   public class NodoArbol { 30 usages new *
41  |     public void setHijoDerecho(NodoArbol hijoDerecho) { no usages new *
42  |         this.hijoDerecho = hijoDerecho;
43  |
44
45     @Override new *
46  ©†     public boolean equals(Object object) {
47     |     if (this == object) return true;
48     |     if (object == null || getClass() != object.getClass()) return false;
49     |     NodoArbol nodoArbol = (NodoArbol) object;
50     |     return Objects.equals(dato, nodoArbol.dato);
51     }
52
53     @Override new *
54  ©†     public int hashCode() {
55     |     return Objects.hash(dato);
56     }
57
58     @Override new *
59  ©†     public String toString() {
60     |     return "NodoArbol{" +
61     |         "dato=" + dato +
62     |         ", hijoIzquierdo=" + hijoIzquierdo +
63     |         ", hijoDerecho=" + hijoDerecho +
64     |         '}';
65     }
66
67 }
```

[Home](#)[About](#)[Content](#)[Others](#)

>Main.java × NodoArbol.java

```
1 ► public class Main {  
2 ►     public static void main(String[] args) {  
3         // Árbol 1  
4         NodoArbol nodo1 = new NodoArbol( dato: 1);  
5         NodoArbol nodo5 = new NodoArbol( dato: 5, nodo1, hijoDerecho: null);  
6         NodoArbol nodo25 = new NodoArbol( dato: 25);  
7         NodoArbol nodo15 = new NodoArbol( dato: 15, hijolquierdo: null, nodo25);  
8         NodoArbol raiz1 = new NodoArbol( dato: 10, nodo5, nodo15);  
9  
10        System.out.println("Árbol 1:");  
11        System.out.println(raiz1);  
12  
13        // Árbol 2  
14        NodoArbol nodoSusan = new NodoArbol( dato: "Susan");  
15        NodoArbol nodoDiana = new NodoArbol( dato: "Diana");  
16        NodoArbol nodoPedro = new NodoArbol( dato: "Pedro", nodoSusan, nodoDiana);  
17        NodoArbol nodoMario = new NodoArbol( dato: "Mario");  
18        NodoArbol raiz2 = new NodoArbol( dato: "Diego", nodoPedro, nodoMario);  
19  
20        System.out.println("\nÁrbol 2:");  
21        System.out.println(raiz2);  
22    }  
23
```



# Home

# About

# Content

## Others

Run Main

```
C:\Users\rodri\.jdks\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.0.2\lib\idea_rt.jar=50758:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.0.2\bin" -Dfile.encoding=UTF-8 -classpath "C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.0.2\lib\idea_rt.jar;C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.0.2\bin" Main
Árbol 1:
NodoArbol{dato=10, hijoIzquierdo=NodoArbol{dato=5, hijoIzquierdo=NodoArbol{dato=1, hijoIzquierdo=null, hijoDerecho=null}, hijoDerecho=null}, hijoDerecho=null}

Árbol 2:
NodoArbol{dato=Diego, hijoIzquierdo=NodoArbol{dato=Pedro, hijoIzquierdo=NodoArbol{dato=Susan, hijoIzquierdo=null, hijoDerecho=null}, hijoDerecho=NodoArbol{dato=Juan, hijoIzquierdo=null, hijoDerecho=null}}}

Process finished with exit code 0
```

[Home](#)[About](#)[Content](#)[Others](#)

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Structure:** Shows files `Main.java` and `NodoArbol.java`.
- Run Configuration:** The configuration `Main` is selected.
- Output Log:** Displays the command used to run the application: `java -jar C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.0.2\bin" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 Main`. Below this, the Java code for a binary search tree node is shown, indicating the application is running correctly.
- Bottom Status Bar:** Shows the current file path (`doArbol > src > Main`), file encoding (23:2 LF), character set (UTF-8), and code style settings (4 spaces).