

EJEMPLO

```
import javax.swing.*;
import javax.swing.tree.DefaultMutableTreeNode;
import java.awt.*;

public class DemoJTreeCorto {
    public static void main(String[] args) {
        SwingUtilities.invokeLater(() -> {
            // Nodo raíz
            DefaultMutableTreeNode root = new DefaultMutableTreeNode("Universidad");

            // Facultades
            DefaultMutableTreeNode ing = new DefaultMutableTreeNode("Ingeniería");
            DefaultMutableTreeNode hum = new DefaultMutableTreeNode("Humanidades");

            // Agregar facultades a la raíz
            root.add(ing);
            root.add(hum);

            // Ciclos para Ingeniería
            ing.add(new DefaultMutableTreeNode("Ciclo 1 - 2025"));
            ing.add(new DefaultMutableTreeNode("Ciclo 2 - 2025"));

            // Ciclos para Humanidades
            hum.add(new DefaultMutableTreeNode("Ciclo 1 - 2025"));
            hum.add(new DefaultMutableTreeNode("Ciclo 2 - 2025"));

            // Crear el árbol con la raíz
            JTree tree = new JTree(root);

            // Mostrar en ventana
            JFrame frame = new JFrame("Ejemplo JTree - Universidad");
            frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
            frame.add(new JScrollPane(tree), BorderLayout.CENTER);
            frame.setSize(300, 300);
            frame.setLocationRelativeTo(null);
            frame.setVisible(true);
        });
    }
}
```

1) Estructura general y clases importadas

```
import javax.swing.*;
import javax.swing.tree.DefaultMutableTreeNode;
import java.awt.*;
```

- `javax.swing.*`: trae las clases de Swing (GUI) como `JFrame`, `JTree`, `JScrollPane`, etc.
- `javax.swing.tree.DefaultMutableTreeNode`: es la clase que representa **nodos** de un árbol (puedes agregar/quitar hijos).
- `java.awt.*`: utilidades de layout y contenedores (por ejemplo `BorderLayout`).

2) Punto de entrada y EDT (Event Dispatch Thread)

```
public class DemoJTreeCorto {
    public static void main(String[] args) {
        SwingUtilities.invokeLater(() -> {
            // ...
        });
    }
}
```

- `main` es el inicio del programa.
- `SwingUtilities.invokeLater(...)`: ejecuta el código de la interfaz **en el hilo de eventos de Swing (EDT)**.

Esto es buena práctica: todo acceso/creación de componentes gráficos debe hacerse en el EDT para evitar condiciones de carrera o comportamientos raros.

3) Crear la raíz del árbol

```
DefaultMutableTreeNode root = new DefaultMutableTreeNode("Universidad");
```

- Crea el **nodo raíz** del árbol.
- El texto "Universidad" es el **userObject** del nodo (lo que verá el `JTree` al pintarlo, usando `toString()`).

4) Crear nodos de primer nivel (hijos de la raíz)

```
DefaultMutableTreeNode ing = new DefaultMutableTreeNode("Ingeniería");
DefaultMutableTreeNode hum = new DefaultMutableTreeNode("Humanidades");
```

- Dos nodos que representan las facultades.
- Aún no están "colgados" en el árbol; solo están creados.

5) Armar la jerarquía raíz → facultades

```
root.add(ing);
root.add(hum);
```

- Ahora sí, **haces a ing e hum hijos de root**.
- El orden en que los agregas define el orden de visualización (salvo que uses un `sorter`).

6) Agregar subnodos (ciclos) a cada facultad

```
ing.add(new DefaultMutableTreeNode("Ciclo 1 - 2025"));
ing.add(new DefaultMutableTreeNode("Ciclo 2 - 2025"));
```

```
hum.add(new DefaultMutableTreeNode("Ciclo 1 - 2025"));
hum.add(new DefaultMutableTreeNode("Ciclo 2 - 2025"));
```

- A **Ingeniería**: le cuelgas dos **hijos** con esos textos.
- A **Humanidades**: lo mismo.
- Estos nodos (ciclos) son, en esta estructura, **hojas** (no tienen hijos).

Hasta aquí, la estructura **lógica** queda:

```
Universidad
├── Ingeniería
│   ├── Ciclo 1 - 2025
│   └── Ciclo 2 - 2025
└── Humanidades
    ├── Ciclo 1 - 2025
    └── Ciclo 2 - 2025
```

7) Crear el componente visual del árbol

```
JTree tree = new JTree(root);
```

- Crea un JTree que **pinta** el árbol a partir del nodo raíz root.
- Internamente generará un TreeModel por defecto (si necesitaras mutaciones complejas en tiempo de ejecución, usarías un DefaultTreeModel y notificaciones como nodesWereInserted, etc., pero para este ejemplo no hace falta).

8) Preparar la ventana y mostrar el árbol

```
JFrame frame = new JFrame("Ejemplo JTree - Universidad");
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
frame.add(new JScrollPane(tree), BorderLayout.CENTER);
frame.setSize(300, 300);
frame.setLocationRelativeTo(null);
frame.setVisible(true);
```

- JFrame: ventana principal con el título indicado.
- setDefaultCloseOperation(EXIT_ON_CLOSE): al cerrar la ventana, la app termina.
- add(new JScrollPane(tree), BorderLayout.CENTER):
 - Envuelve el JTree en un JScrollPane para tener **barras de desplazamiento** si el contenido crece.
 - Lo ubica en el centro del layout por defecto del JFrame (BorderLayout).
- setSize(300, 300): tamaño de la ventana (en píxeles).
- setLocationRelativeTo(null): centra la ventana en la pantalla.
- setVisible(true): finalmente, **muestra** la ventana.