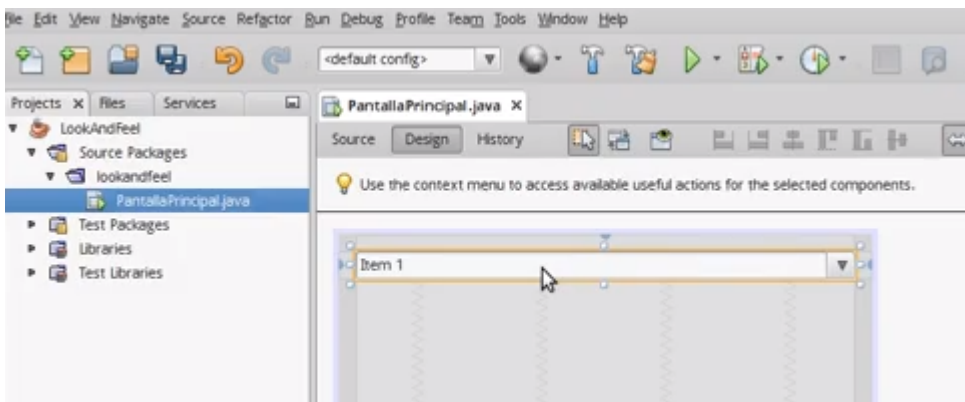


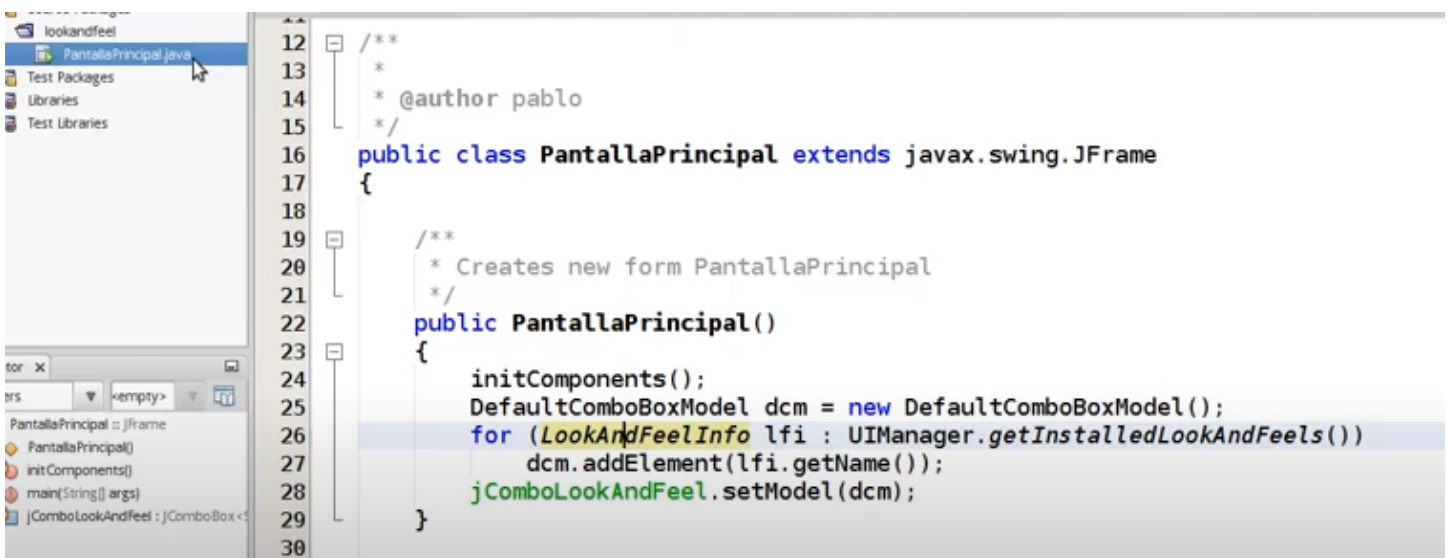
Trabajando la apariencia de los componentes. Look and Feel Swing.

Vamos a trabajar la apariencia de un combobox, como sabemos es un componente para seleccionar opciones, este ejemplo te mostrará algunos principios básicos para aportar usabilidad a nuestras aplicaciones con look and feel, no solo estética o apariencia es lo que cuenta, así que practicaremos un poco con este ejemplo. Antes que nada revisa primero esta [INTRO para que comprendas lo que es un Look and Feel](#).

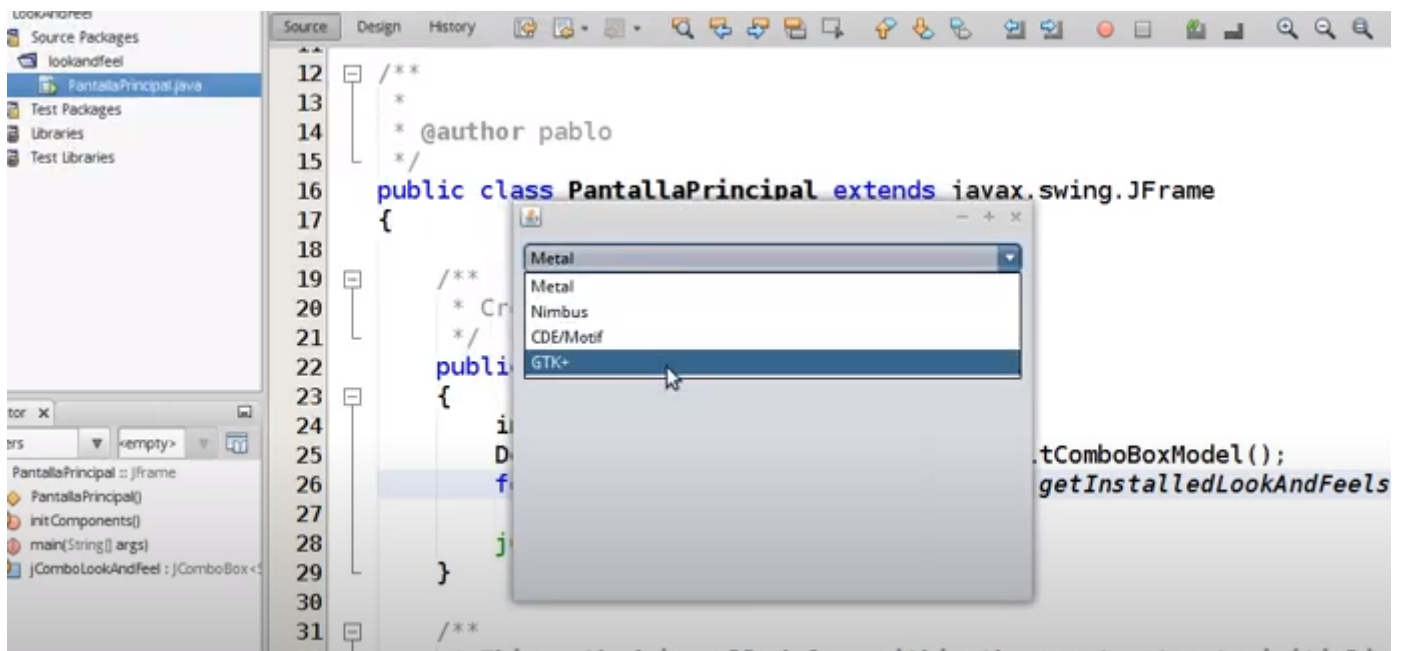
Vamos a crear un nuevo proyecto y a integrar el combobox con nombre de variable JComboLookAndFeel



Vamos a hacer que nuestro combo nos muestre cuantos look and feels tengo instalados en mi sistema agregando unas líneas de código en la inicialización de componentes para consultar en la clase UIManager que es donde se alojan los Look and Feels, por medio de un for, cuales son los que tengo instalados por medio de un método que devuelva la lista y asignando el modelo al combo

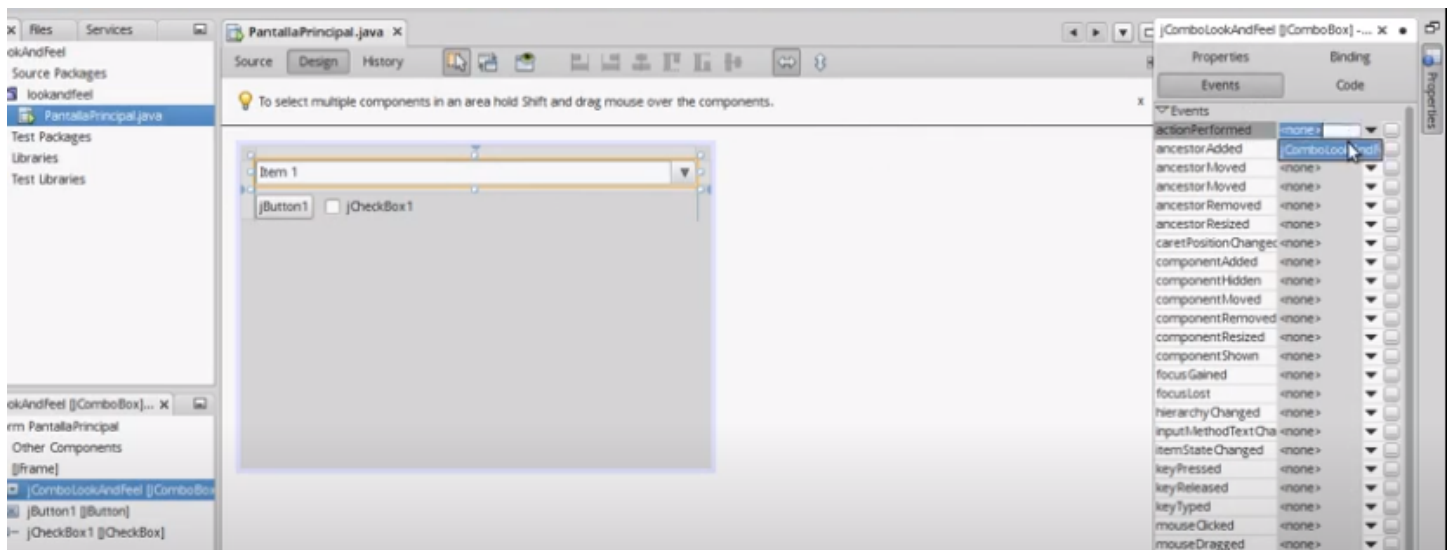


Ejecutamos y vemos el primer resultado

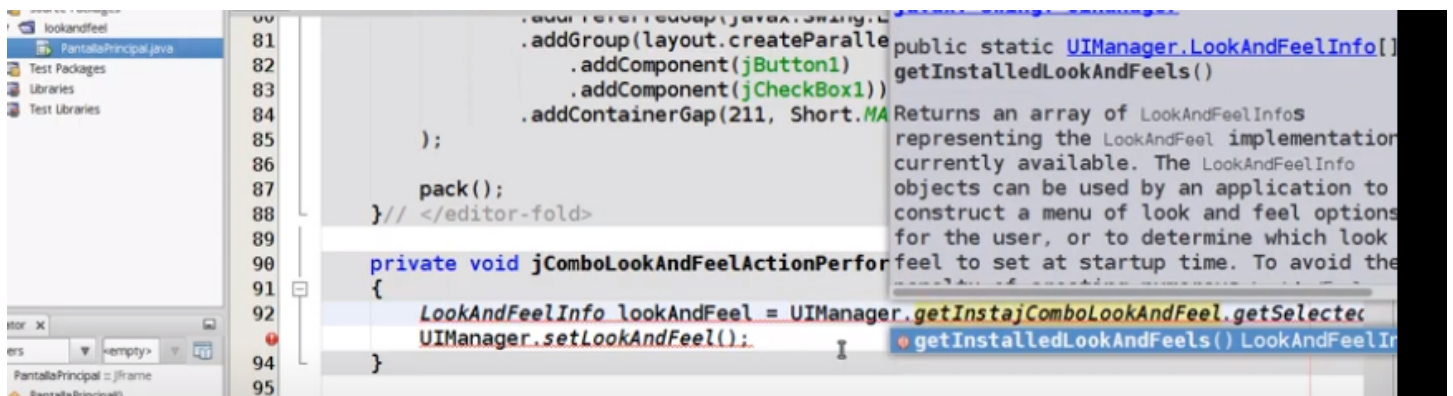


El que tenemos por defecto es el Nimbus, pero ahora vamos a hacer que cada vez que seleccione alguno, la apariencia cambie

Agreguemos unos componentes más, como un botón y un checkbox, además programaremos el actionPerformed en los eventos del combo



Volveremos a usar el UIManager, lo haremos con varias líneas para dejarlo claro, pero podríamos hacerlo en una línea ¿te atreves?



La línea completa es:

```
LookAndFeelInfo lookAndFeel =  
UIManager.getInstalledLookAndFeels()[jComboLookAndFeel.getSelectedIndex()];
```

El código nos obliga a usar una excepción...

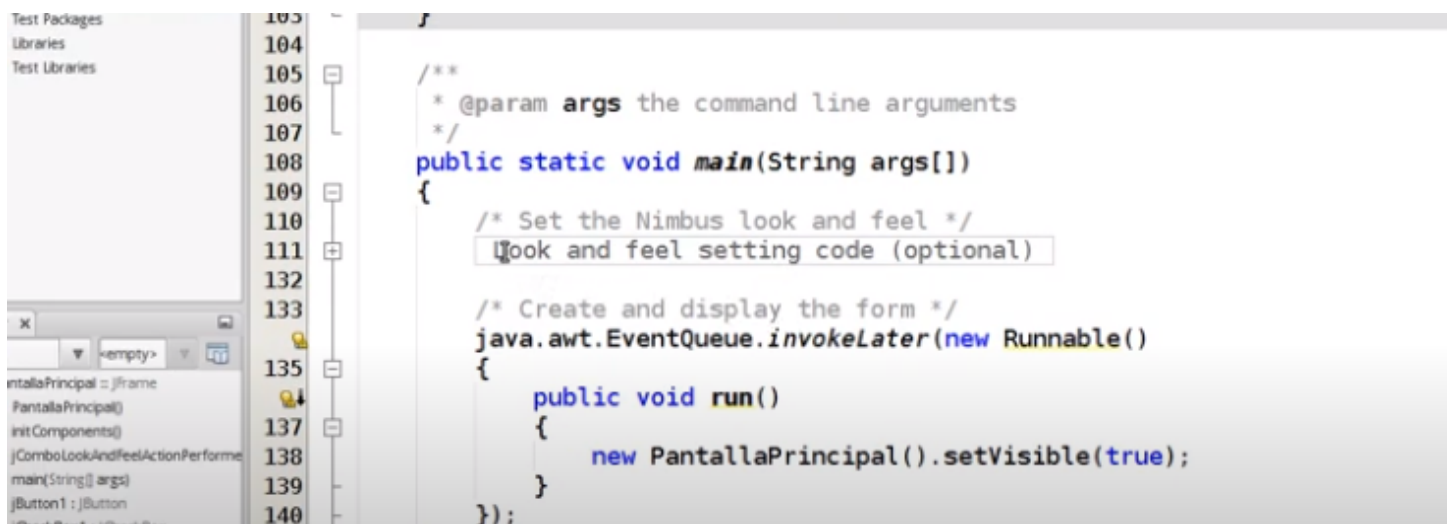
```
private void jComboLookAndFeelActionPerformed(java.awt.event.ActionEvent evt)  
{  
    try  
    {  
        LookAndFeelInfo lookAndFeel = UIManager.getInstalledLookAndFeels()[jComb  
        UIManager.setLookAndFeel(lookAndFeel.getClassName());  
    }  
    catch (Throwable e)  
    {  
    }  
}
```

Con esta línea accedo a la posición seleccionada en el combo. Agregaremos una línea para refrescar el árbol de componentes del frame principal con el método `updateComponent`

```
private void jComboLookAndFeelActionPerformed(java.awt.event.Ac  
{  
    try  
    {  
        LookAndFeelInfo lookAndFeel = UIManager.getInstalledLoc  
        UIManager.setLookAndFeel(lookAndFeel.getClassName());  
        SwingUtilities.updateComponentTreeUI(this);  
    }  
    catch (Throwable e)  
    {  
    }  
}
```

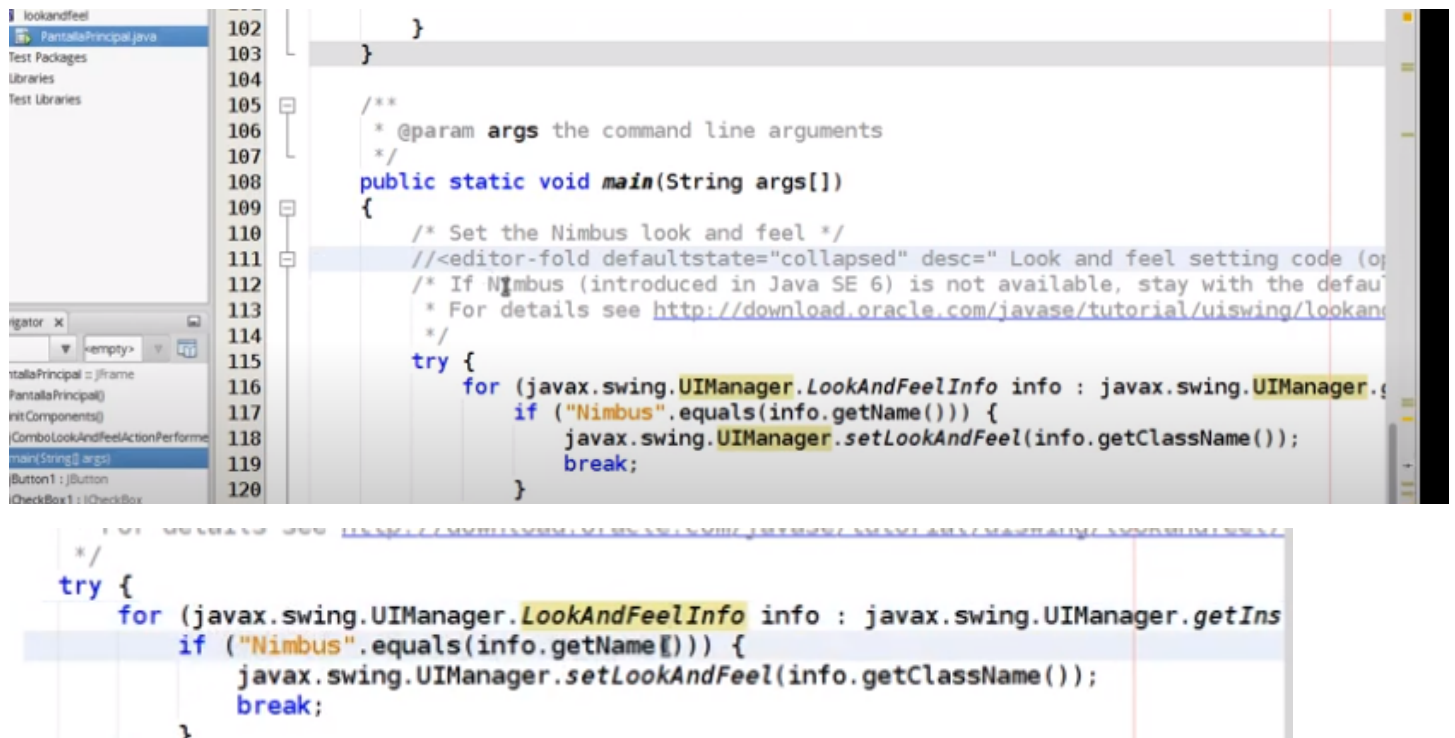
Ahora ejecuta y deberás ver los cambios cada vez que selecciones un nuevo LookAndFeel,

¿Por qué se abre por defecto el Nimbus? puedes comprobarlo en el main de la clase



```
/**  
 * @param args the command line arguments  
 */  
public static void main(String args[])  
{  
    /* Set the Nimbus look and feel */  
    /* Look and feel setting code (optional) */  
  
    /* Create and display the form */  
    java.awt.EventQueue.invokeLater(new Runnable()  
    {  
        public void run()  
        {  
            new PantallaPrincipal().setVisible(true);  
        }  
    });  
}
```

Abriendo el código donde dice Look And Feel Setting code, ahí podrás visualizar como se utiliza de manera similar a como nosotros lo hemos hecho



```
102 }
103 }
104
105 /**
106  * @param args the command line arguments
107  */
108 public static void main(String args[])
109 {
110     /* Set the Nimbus look and feel */
111     //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional)
112     /* If Nimbus (introduced in Java SE 6) is not available, stay with the default
113      * For details see http://download.oracle.com/javase/7/tutorial/ui/lookandfeel/
114      */
115     try {
116         for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
117             if ("Nimbus".equals(info.getName())) {
118                 javax.swing.UIManager.setLookAndFeel(info.getClassName());
119                 break;
120             }
121         }
122     } catch (ClassNotFoundException ex) {
123         java.util.logging.Logger.getLogger(Main.class).log(java.util.logging.Level.SEVERE, null, ex);
124     } catch (InstantiationException ex) {
125         java.util.logging.Logger.getLogger(Main.class).log(java.util.logging.Level.SEVERE, null, ex);
126     } catch (IllegalAccessException ex) {
127         java.util.logging.Logger.getLogger(Main.class).log(java.util.logging.Level.SEVERE, null, ex);
128     }
129 }
```

En ese código, revisa la lista y si está el NIMBUS instalado lo usa por defecto.

Ahora es preciso que amplíes tus conocimientos sobre la lección dada, investiga las apariencias del resto de look and feels instalados, verifica como estos pueden hacer que la apariencia cambie y le brinde comodidad al usuario.

Investiga imágenes de cada uno de estos Look and Feels y compártelas enviándome algunos enlaces.

Prueba el siguiente Look And Feel y muestra tu resultado

https://sourceforge.net/projects/napkinlaf/?source=typ_redirect