

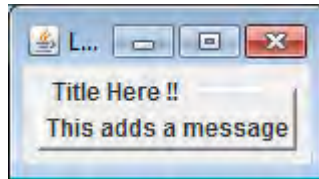


## Borders – createTitledBorder()

---

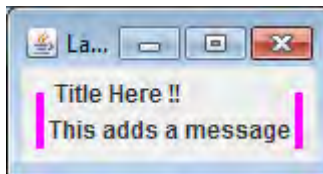
### Raised border - left justified title on top

```
BorderFactory.createTitledBorder(BorderFactory.createRaisedBevelBorder()  
    , "Title Here !!"  
    , TitledBorder.LEFT, TitledBorder.TOP);
```



### Matte border - left justified title below top

```
BorderFactory.createTitledBorder(BorderFactory.createMatteBorder(0,4,0,4  
    , Color.magenta)  
    , "Title Here !!"  
    , TitledBorder.LEFT, TitledBorder.TOP);
```



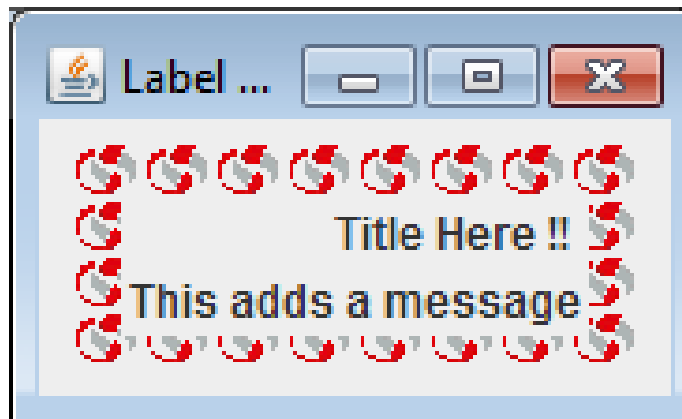
# Borders – createTitledBorder()



## Matte border with icon - right justified title above top

```
ImageIcon icon = new ImageIcon("c:/course/Seericon.gif");
```

```
BorderFactory.createTitledBorder(BorderFactory.createMatteBorder(15,15,13,15,icon)  
    ,"Title Here !!"  
    ,TitledBorder.RIGHT,TitledBorder.BELOW_TOP);
```

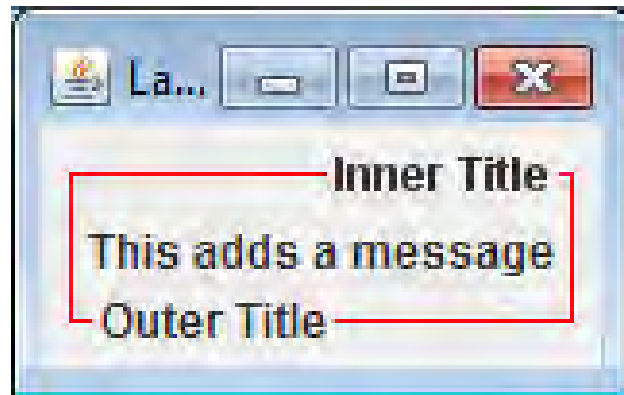




## Borders – Nested createTitledBorder()

This next example shows the result of using a TitledBorder inside another TitledBorder (ideal when two titles are needed).

```
BorderFactory.createTitledBorder(  
    BorderFactory.createTitledBorder(BorderFactory.createLineBorder(Color.red)  
        , "Inner Title"  
        , TitledBorder.RIGHT, TitledBorder.TOP)  
    , "Outer Title"  
    , TitledBorder.LEFT, TitledBorder.BOTTOM);
```



This technique though can get confusing if the nesting continues.



## Borders – Fonts with createTitledBorder()

---

There are two further arguments that can be added to a TitledBorder, font and colour of the title.

Remembering that any font can be used by the Developer providing it has been installed on the machine running the process ...

```
BorderFactory.createTitledBorder(BorderFactory.createRaisedBevelBorder()  
    , "Title Here !!"  
    , TitledBorder.LEFT, TitledBorder.TOP  
    , new Font("Allegro BT", Font.BOLD, 18)  
    , Color.red);
```

