

JButton Widget

```
public class ButtonExample extends JFrame implements ActionListener {
```

```
...
JPanel titlePanel, scorePanel, buttonPanel;
JLabel redLabel, blueLabel, redScore, blueScore;
JButton redButton, blueButton, resetButton;
...
```

```
private void initGUI () {
```

```
...
    redButton = new JButton("Red Score!");
    redButton.setLocation(0, 0);
    redButton.setSize(120, 30);
    redButton.addActionListener(this);
    buttonPanel.add(redButton);

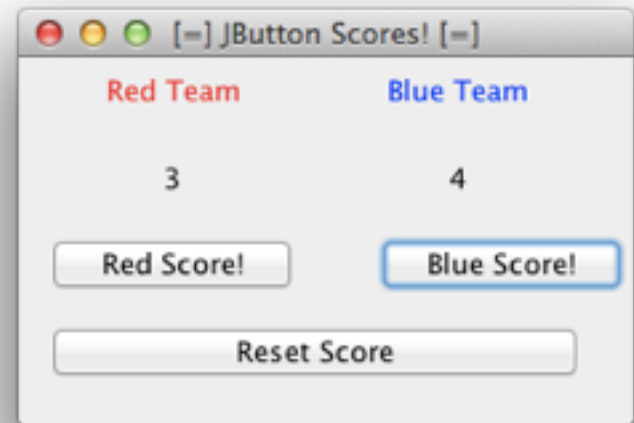
```

```
...
}
```

```
public void actionPerformed(ActionEvent e) {
    if ( e.getSource() == redButton ) {
        redScoreAmount = redScoreAmount + 1;
        redScore.setText(""+redScoreAmount);
    } else ...

```

```
    }
}
```



see: `example.swing.misc.ButtonExample`

JButton Widget

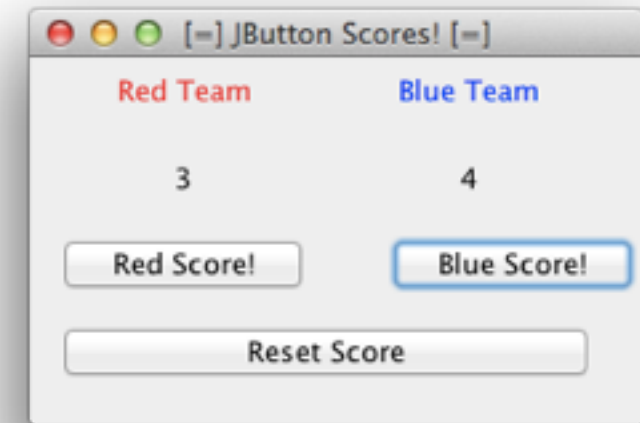
```
public class ButtonExample extends JFrame implements ActionListener {
```

```
...
JPanel titlePanel, scorePanel, buttonPanel;
JLabel redLabel, blueLabel, redScore, blueScore;
JButton redButton, blueButton, resetButton;
...
```

```
private void initGUI () {
```

```
...
    redButton = new JButton("Red Score!");
    redButton.setLocation(0, 0);
    redButton.setSize(120, 30);
    redButton.addActionListener(this);
    buttonPanel.add(redButton);
    ...
}
```

```
public void actionPerformed(ActionEvent e) {
    if ( e.getSource() == redButton ) {
        redScoreAmount = redScoreAmount + 1;
        redScore.setText(""+redScoreAmount);
    } else ...
}
}
```



crear un botón con el
text "Red Score!"

see: [example.swing.misc.ButtonExample](#)

JButton Widget

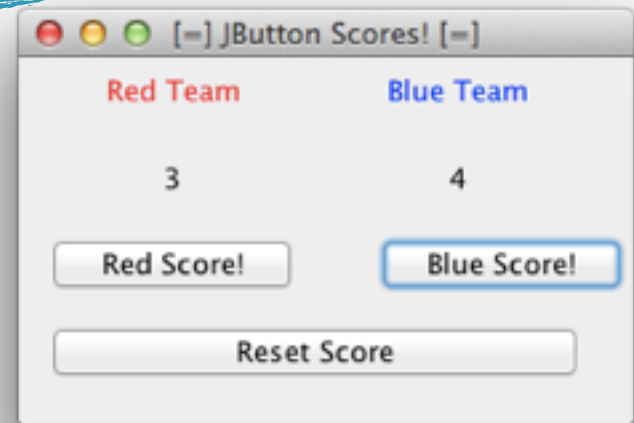
```
public class ButtonExample extends JFrame implements ActionListener {
```

```
...
JPanel titlePanel, scorePanel, buttonPanel;
JLabel redLabel, blueLabel, redScore, blueScore;
JButton redButton, blueButton, resetButton;
...
```

```
private void initGUI () {
```

```
...
    redButton = new JButton("Red Score!");
    redButton.setLocation(0, 0);
    redButton.setSize(120, 30);
    redButton.addActionListener(this);
    buttonPanel.add(redButton);
    ...
}
```

```
public void actionPerformed(ActionEvent e) {
    if ( e.getSource() == redButton ) {
        redScoreAmount = redScoreAmount + 1;
        redScore.setText(""+redScoreAmount);
    } else ...
}
```



crear un botón con el text "Red Score!"

añadir un oyente (listener) para llamarle cuando se hace click sobre el botón

see: [example.swing.misc.ButtonExample](#)

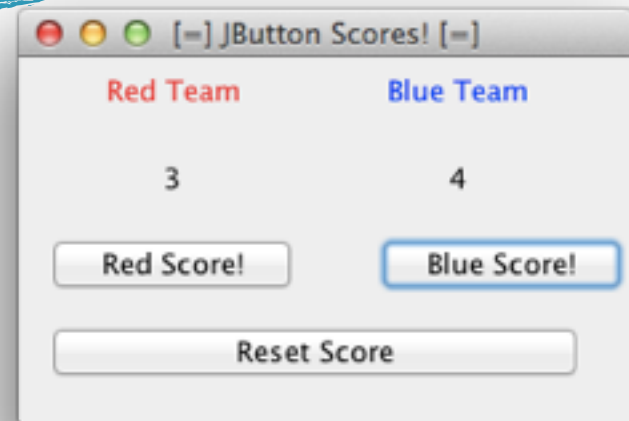
JButton Widget

```
public class ButtonExample extends JFrame implements ActionListener {
```

```
...
JPanel titlePanel, scorePanel, buttonPanel;
JLabel redLabel, blueLabel, redScore, blueScore;
JButton redButton, blueButton, resetButton;
...
```

```
private void initGUI () {
    ...
    redButton = new JButton("Red Score!");
    redButton.setLocation(0, 0);
    redButton.setSize(120, 30);
    redButton.addActionListener(this);
    buttonPanel.add(redButton);
    ...
}
```

```
public void actionPerformed(ActionEvent e) {
    if ( e.getSource() == redButton ) {
        redScoreAmount = redScoreAmount + 1;
        redScore.setText(""+redScoreAmount);
    } else ...
}
```



Los widgets son atributos,
para poder cambiar sus
propiedades, etc.

crear un botón con el
text "Red Score!"

añadir un oyente (listener)
para llamarle cuando se
hace click sobre el botón

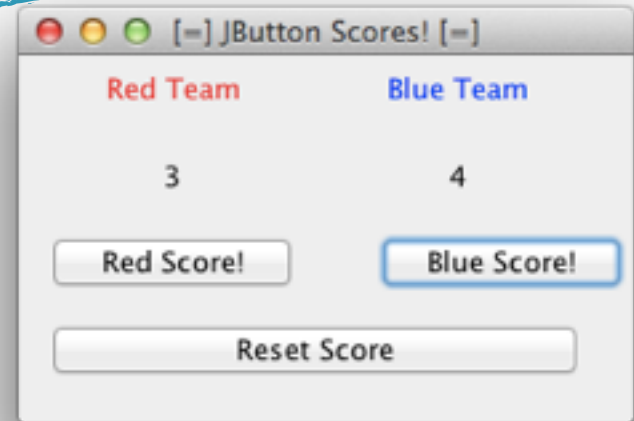
JButton Widget

```
public class ButtonExample extends JFrame implements ActionListener {
```

```
...
JPanel titlePanel, scorePanel, buttonPanel;
JLabel redLabel, blueLabel, redScore, blueScore;
JButton redButton, blueButton, resetButton;
...
```

```
private void initGUI () {
    ...
    redButton = new JButton("Red Score!");
    redButton.setLocation(0, 0);
    redButton.setSize(120, 30);
    redButton.addActionListener(this);
    buttonPanel.add(redButton);
    ...
}
```

```
public void actionPerformed(ActionEvent e) {
    if (e.getSource() == redButton) {
        redScoreAmount = redScoreAmount + 1;
        redScore.setText(""+redScoreAmount);
    } else ...
}
```



Los widgets son atributos, para poder cambiar sus propiedades, etc.

crear un botón con el text "Red Score!"

añadir un oyente (listener) para llamarle cuando se hace click sobre el botón

usa `e.getSource()` para saber de donde viene el evento

Usa Clases Anónimas para Listeners

```
public class ButtonExample extends JFrame {  
    ...  
    JPanel titlePanel, scorePanel, buttonPanel;  
    JLabel redLabel, blueLabel, redScore, blueScore;  
    JButton redButton, blueButton, resetButton;  
    ...  
    private void initGUI () {  
        ...  
        redButton = new JButton("Red Score!");  
        redButton.setLocation(0, 0);  
        redButton.setSize(120, 30);  
        redButton.addActionListener( new ActionListener() {  
            public void actionPerformed(ActionEvent e) {  
                redScoreAmount = redScoreAmount + 1;  
                redScore.setText("" + redScoreAmount);  
            }  
        });  
        buttonPanel.add(redButton);  
        ...  
    }  
    ...  
}
```

see: [example.swing.misc.ButtonExample_1](#)

Usa Clases Anónimas para Listeners

```
public class ButtonExample extends JFrame {  
    ...  
    JPanel titlePanel, scorePanel, buttonPanel;  
    JLabel redLabel, blueLabel, redScore, blueScore;  
    JButton redButton, blueButton, resetButton;  
    ...  
    private void initGUI () {  
        ...  
        redButton = new JButton("Red Score!");  
        redButton.setLocation(0, 0);  
        redButton.setSize(120, 30);  
        redButton.addActionListener( new ActionListener() {  
            public void actionPerformed(ActionEvent e) {  
                redScoreAmount = redScoreAmount + 1;  
                redScore.setText("" + redScoreAmount);  
            }  
        });  
        buttonPanel.add(redButton);  
        ...  
    }  
    ...  
}
```

Cada botón tiene su "listener", no hace falta usar e.source()

Usa Clases Anónimas para Listeners

```
public class ButtonExample extends JFrame {  
    ...  
    JPanel titlePanel, scorePanel, buttonPanel;  
    JLabel redLabel, blueLabel, redScore, blueScore;  
    JButton redButton, blueButton, resetButton;  
    ...  
    private void initGUI () {  
        ...  
        redButton = new JButton("Red Score!");  
        redButton.setLocation(0, 0);  
        redButton.setSize(120, 30);  
        redButton.addActionListener( new ActionListener() {  
            public void actionPerformed(ActionEvent e) {  
                redScoreAmount = redScoreAmount + 1;  
                redScore.setText("" + redScoreAmount);  
            }  
        });  
        buttonPanel.add(redButton);  
        ...  
    }  
    ...  
}
```

La clase ButtonExample no implementa la interfaz ActionListener, because it does not listen to any JButton.

Cada botón tiene su "listener", no hace falta usar e.source()

JButton con Icon

```
private void initGUI() {
```

```
...
```

```
exitButton = new JButton("Exit");
```

```
exitButton.setLocation(0, 0);
```

```
exitButton.setIcon( new ImageIcon("src/tp/examples/swing/icons/exit.png") );
```

```
mainPanel.add(exitButton);
```

```
exitButton = new JButton("Undo");
```

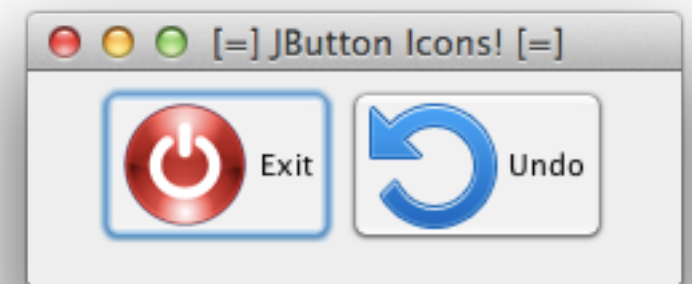
```
exitButton.setLocation(0, 0);
```

```
exitButton.setIcon( new ImageIcon( IconsDir.class.getResource("undo.png") ) );
```

```
mainPanel.add(exitButton);
```

```
...
```

```
}
```



ruta absoluta o relativa a la raíz del proyecto ...

relativa al paquete (carpeta) en el que está la clase IconsDir