

Gestion d'évènements par délégation

Principe :

Le mécanisme de délégation pour la gestion des événements consiste à indiquer qu'un objet graphique est écouté par un objet «écouteur» d'une autre classe. Cet objet «écouteur» lancera une méthode convenue par Java lors de l'action spécifique de l'utilisateur sur cet objet graphique.

Exemple :

```
import java.awt.*;
import java.awt.event.*;

// ...

Button bt = new Button("Mon Bouton");
EcouteurBouton ecouteur = new EcouteurBouton();
bt.addActionListener(ecouteur); // l'ecouteur est associe au bouton

// ...

class EcouteurBouton implements ActionListener {

    public void actionPerformed(ActionEvent e) {
        System.out.println("Coucou");
    }
}
```

- Crée une fenêtre **awt** de type «frame» contenant trois boutons :
 - ★ Couleur : pour modifier la couleur :
 - soit du fond du bouton lorsque le pointeur est sur le bouton
 - soit du fond de la fenêtre lors d'un clic sur le bouton.
 - ★ Fenêtre : pour ouvrir une autre fenêtre contenant divers composants, au choix.
 - ★ Quitter pour quitter l'application
- Idem avec Swing.