

Création d'une interface graphique

Ce document présente un exemple simple d'interface graphique servant d'introduction à la création d'interfaces graphiques.

Copier la classe DemoFrame dans Eclipse et exécuter celle-ci pour comprendre la fonction de l'application (manipulation de chaîne de caractères).

Consulter ensuite les pages suivantes pour voir quels sont les attributs utilisés et la composition de l'application.

Structure de l'application

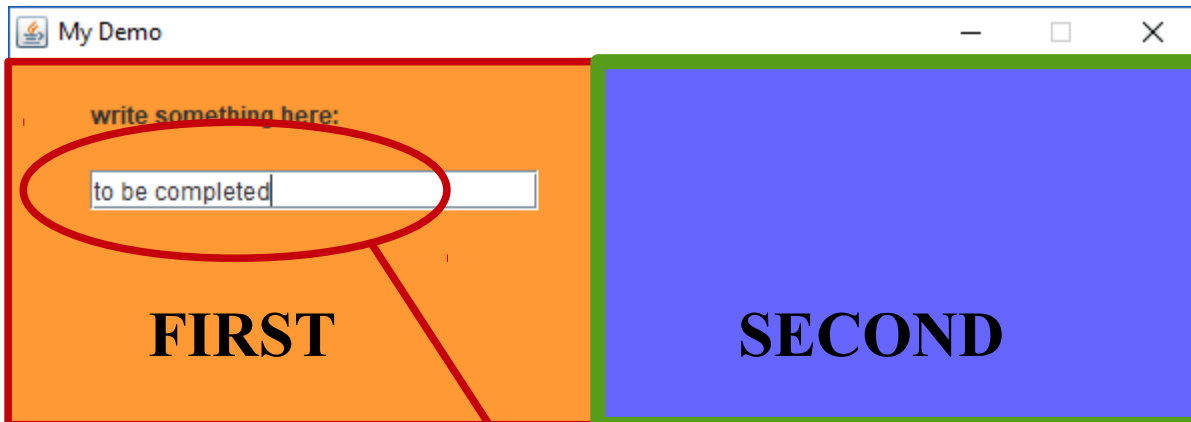


Code pour les deux panneaux (constructeur de DemoFrame)

top et **bottom** sont deux attributs de la classe

```
public DemoFrame(String arg0) {  
    super(arg0);  
    setResizable(false);  
    Container pane=getContentPane();  
    // Layout : une grille de deux lignes égales pour top et bottom  
    pane.setLayout(new GridLayout(2,0));  
    top=new JPanel();  
    top.setBackground(new Color(0xFF9933));  
    bottom=new JPanel();  
    bottom.setBackground(new Color(0xFFB266));  
    pane.add(top);  
    pane.add(bottom);  
}
```

Composants du panneau top



text1 : JTextField

Composants du panneau top

Le panneau top contient deux sous-panneaux **first** et **second**. Le panneau second est inutilisé : il servira d'exercice en fin de tutoriel.

On utilise à nouveau le layout GridLayout pour placer les sous-panneaux en deux colonnes.

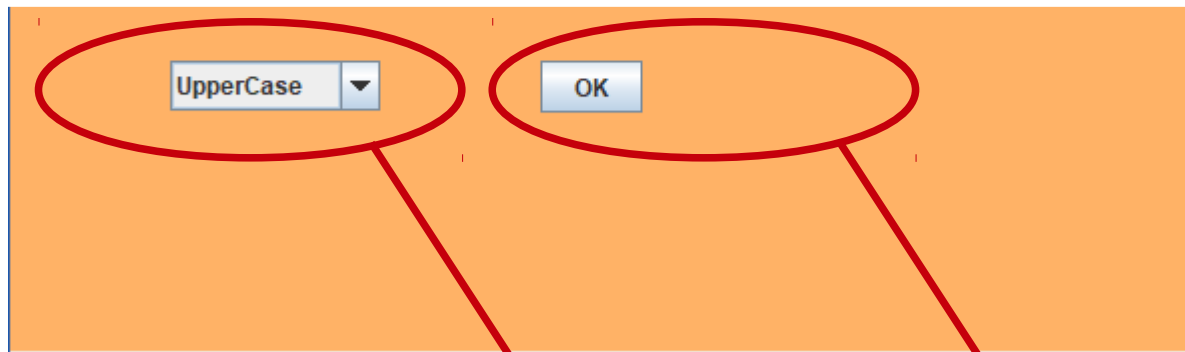
Le panneau **first** contient un label pour indiquer la fonction de l'élément **text1**. Cet élément est une zone de texte et est un attribut de la classe DemoFrame. Noter le layout de type FlowLayout qui place les éléments de haut en bas et de gauche à droite selon la place disponible.

Le code correspondant est indiqué sur la diapo suivante.

Code du panneau top

```
public void completeTop(){
    top.setLayout(new GridLayout(0,2));
    JPanel first=new JPanel();
    first.setBackground(new Color(0xFF9933));
    first.setLayout(new FlowLayout(FlowLayout.LEADING,40,20));
    first.add(new JLabel("write something here:"));
    text1=new JTextField("to be completed");
    text1.setColumns(20);
    first.add(text1);
    JPanel second=new JPanel();
    second.setBackground(new Color(0x6666FF));
    top.add(first);
    top.add(second);
}
```

Composants du panneau bottom



box: JComboBox ok : JButton

Composants du panneau bottom

Le panneau contient une boîte combinée (**comboBox**) et un bouton **OK** (attributs).

Ces éléments sont placés de gauche à droite par le layout FlowLayout.

Code :

```
public void completeBottom() {  
    bottom.setLayout(new  
FlowLayout(FlowLayout.LEADING,80,40));  
    box = new JComboBox(legend);  
    bottom.add(box);  
    ok=new JButton("OK");  
    bottom.add(ok);  
    ok.addActionListener(this);  
}
```


Gestion des événements

Noter l'instruction suivante :

ok.addActionListener(this);

Elle signifie que le bouton OK est contrôlé par **this**, qui est la frame elle-même (application DemoFrame).

Cela est rendu possible grâce à la déclaration de DicoFrame :

```
public class DemoFrame extends JFrame implements ActionListener
```

L'interface **ActionListener** définit une méthode **actionPerformed** qui réagit à l'action de l'utilisateur sur le bouton OK.

Contrôle de l'événement

```
public void actionPerformed(ActionEvent evt) {  
    String event=evt.getActionCommand();  
    int index=box.getSelectedIndex();  
    if (event.equals("OK")){  
        String input=text1.getText();  
        if (legend[index].equals("UpperCase")){  
            text1.setText(input.toUpperCase());  
        }  
    }  
}
```

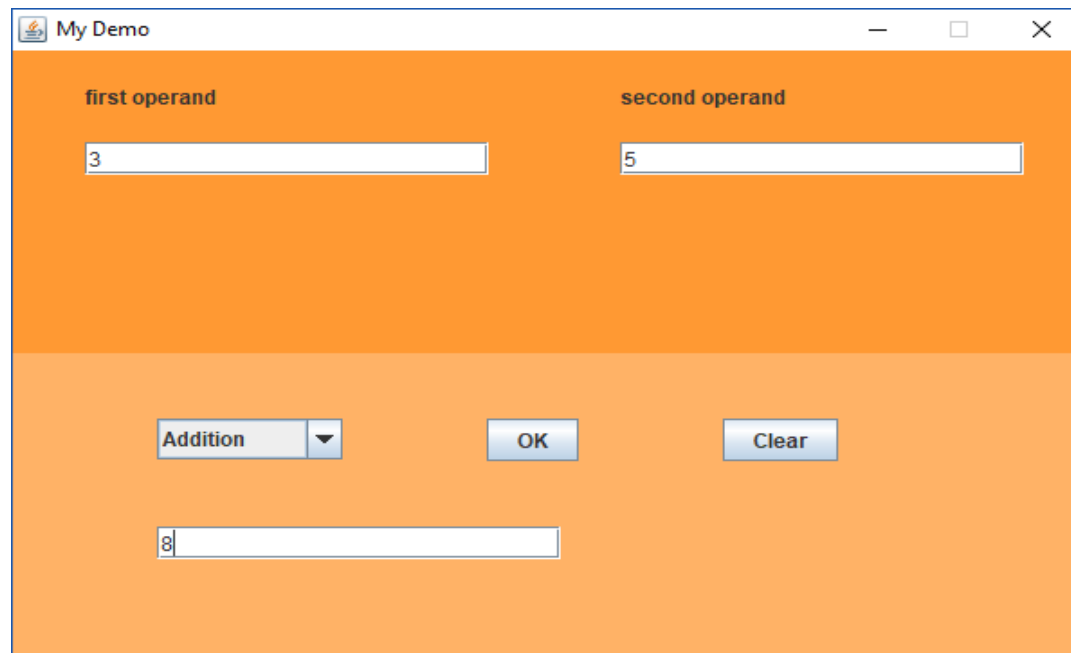
...

Le label de l'événement est récupéré (**event**). Cela permet de vérifier si l'événement provient du bouton OK. La chaîne input récupère ensuite le texte de l'utilisateur et si le choix de l'utilisateur est UpperCase, alors le texte est modifié en ce sens.

Exercice

On vous demande de s'inspirer du code précédent pour écrire une application Calculatrice. Celle-ci comporte deux zones de texte pour les deux opérandes, une boîte combinée pour choisir une opération arithmétique (addition, soustraction, multiplication, division), ainsi que deux boutons :

- un bouton ok pour lancer le calcul. Le résultat sera présenté dans une troisième zone de texte.
- un bouton clear qui efface toutes les zones de texte.



The screenshot shows a window titled "My Demo" with a light orange background. It contains two input fields at the top labeled "first operand" and "second operand", containing the values "3" and "5" respectively. Below these is a dropdown menu currently showing "Addition". To the right of the dropdown are two buttons labeled "OK" and "Clear". At the bottom, there is a single input field containing the value "8".

Vous pouvez reprendre la structure globale de DemoFrame pour écrire votre propre code.

Le panneau first contient déjà l'équivalent du premier opérande (text1). Il suffit de s'en inspirer pour compléter le panneau second avec un deuxième opérande (ajouter un deuxième attribut text2).

Dans le panneau bottom, il faut ajouter un bouton clear qui efface le contenu des opérandes ainsi qu'une troisième zone de texte pour écrire le résultat des opérations.

Bien entendu, changer également tous les labels, dont notamment le texte dans la liste déroulante qui doit indiquer la nature des opérations arithmétiques.