

# Documentation de la calculatrice en Java

## Introduction

Le programme implémente une calculatrice simple en Java, avec une interface graphique utilisant JFrame et des boutons de contrôle pour effectuer des opérations mathématiques de base. Il permet d'effectuer des opérations telles que l'addition, la soustraction, la multiplication, la division, l'inversion du signe, ainsi que d'ajouter des chiffres, un point décimal, et de manipuler l'entrée.

## Classes et objets

- **calcul** : La classe principale qui implémente l'interface ActionListener et crée l'interface graphique ainsi que les actions liées aux boutons.

```
▼ public class calcul implements ActionListener {
```

## Attributs de la classe calcul

- **Variables de type char et double :**
  - operateur : Le caractère représentant l'opération en cours (addition, soustraction, multiplication, division).
  - num1 et num2 : Les deux nombres à utiliser dans l'opération.
  - résultat : Le résultat de l'opération.
- **Composants graphiques de Swing :**
  - fenetre : L'objet JFrame représentant la fenêtre principale de l'application.
  - panel : Le panneau (JPanel) qui contient les boutons numériques et opérationnels dans un GridLayout.
  - text : Le champ de texte (JTextField) où les entrées et résultats sont affichés.
  - **Tableaux de boutons :**
    - bouton[] : Un tableau de 10 boutons pour les chiffres (0-9).
    - fonctionop[] : Un tableau de 9 boutons pour les opérations, le point, et les fonctions comme "effacer" et "supprimer".

```
char operateur;  
JFrame fenetre;  
JPanel panel ;  
  
JButton addition , soustraction , division , supp , eff , negatif , multi , egal , point;  
  
JButton[] bouton = new JButton[10];  
JButton[] fonctionop = new JButton[9];  
JTextField text ;  
  
double num1 , num2, resultat ;
```

## Constructeur calcul()

Le constructeur initialise l'interface graphique de la calculatrice :

### 1. Fenêtre (JFrame) :

- Crée une fenêtre de taille 400x600 pixels avec le titre "Calculatrice".
- Définit la disposition à null pour un placement absolu des composants.
- Associe l'action de fermeture de la fenêtre à la fermeture de l'application.

### 2. Création des boutons :

- **Boutons de chiffres (0-9)** : Chaque bouton est créé et ajouté à un tableau bouton[]. L'écouteur d'événements ActionListener est associé à chaque bouton.
- **Boutons d'opérations** : Des boutons pour chaque opération (+, -, \*, /, etc.) sont créés et ajoutés au tableau fonctionop[].
- **Autres boutons** : Boutons pour la suppression (SUPP), l'effacement (EFF), le signe négatif (-), le point décimal (.).

```
25     fenetre = new JFrame("Calculatrice ");
26     fenetre.setSize(400,600);
27     fenetre.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
28     fenetre.setLayout(null);
29
30
31
32     addition = new JButton("+");
33     fonctionop[0] = addition;
34
35     soustraction = new JButton("-");
36     fonctionop[1] = soustraction;
37
38     division = new JButton("/");
39     fonctionop[2] = division;
40
41     supp = new JButton("SUPP");
42     fonctionop[3] = supp;
43
44     eff = new JButton("EFF");
45     fonctionop[4] = eff;
46
47     negatif = new JButton("-");
48     fonctionop[5] = negatif;
49
50     multi = new JButton("*");
51     fonctionop[6] = multi;
52
53     egal = new JButton("=");
54     fonctionop[7] = egal;
55
56     point = new JButton(".");
57     fonctionop[8] = point;
58
59     for (int i = 0; i < 9; i++) {
60         fonctionop[i].addActionListener(this);
61     }
62
63
64
65
66     for(int i = 0; i < 10; i++) {
67         bouton[i] = new JButton(String.valueOf(i));
68         bouton[i].addActionListener(this);
```

### 3. Disposition des composants :

- Le champ de texte (text) est ajouté en haut de la fenêtre pour afficher les entrées et résultats.
- Le panneau (panel) est configuré en GridLayout(4, 4) et contient les boutons des chiffres et opérations de manière organisée.

- Les autres boutons (SUPP, EFF, et (-)) sont positionnés sous le panneau principal.
4. **Visibilité de la fenêtre :**
- La fenêtre est rendue visible avec `fenetre.setVisible(true)`.

```
panel.add(bouton[1]);
panel.add(bouton[2]);
panel.add(bouton[3]);
panel.add(addition);
panel.add(bouton[4]);
panel.add(bouton[5]);
panel.add(bouton[6]);
panel.add(soustraction);
panel.add(bouton[7]);
panel.add(bouton[8]);
panel.add(bouton[9]);
panel.add(multi);
panel.add(point);
panel.add(bouton[0]);
panel.add(egal);
panel.add(division);

negatif.setBounds(50,430,100,50);
eff.setBounds(250,430,100,50);
supp.setBounds(150,430,100,50);

fenetre.add(panel);
fenetre.add(negatif);
fenetre.add(supp);
fenetre.add(eff);
fenetre.add(text);

fenetre.setVisible(true);
```

### Méthode `actionPerformed(ActionEvent e)`

La méthode `actionPerformed` est appelée chaque fois qu'un bouton est cliqué. Elle gère les événements associés aux différents boutons.

- Chiffres (0-9) :**
  - Si un bouton numérique est cliqué, le chiffre correspondant est ajouté à l'affichage dans le champ de texte (`text`).
- Suppression d'un caractère (SUPP) :**
  - Si le bouton "SUPP" est cliqué, le dernier caractère du texte affiché est supprimé.
- Effacement complet (EFF) :**
  - Si le bouton "EFF" est cliqué, le champ de texte est vidé.
- \*Opérations de calcul (+, -, , /) :**
  - Lorsque l'utilisateur clique sur un bouton d'opération, la valeur affichée dans le champ de texte (`text`) est convertie en un nombre (avec `Double.parseDouble()`), puis cette valeur est stockée dans `num1`. L'opération en cours est enregistrée dans `operateur`.
- Calcul du résultat (=) :**
  - Lorsque le bouton "=" est cliqué, le programme effectue le calcul entre `num1` et le nombre actuellement affiché dans le champ de texte (`num2`) en fonction de l'opération choisie (`operateur`).
  - Le résultat est affiché dans le champ de texte après l'opération.
- Point décimal (.) :**

- Si le bouton "." est cliqué, un point décimal est ajouté à la fin du nombre affiché dans le champ de texte.

## 7. Inversion du signe ((-)) :

- Lorsque le bouton "(-)" est cliqué, le signe du nombre affiché dans le champ de texte est inversé (multiplié par -1).

## Comportement de la calculatrice

- **Affichage** : Le champ de texte affiche les nombres saisis et le résultat des opérations.
- **Calculs** : La calculatrice effectue les quatre opérations de base : addition, soustraction, multiplication, et division.
- **Fonctionnalités supplémentaires** : L'utilisateur peut ajouter un point décimal et inverser le signe des nombres.

```

public void actionPerformed(ActionEvent e){
    for(int i = 0 ; i<10 ; i++){
        if(e.getSource() == bouton[i]){
            text.setText(text.getText().concat(String.valueOf(i)));
        }
    }

    if (e.getSource() == eff){
        text.setText("");
    }

    if (e.getSource() == supp){
        String tab = text.getText();
        text.setText("");
        for (int i = 0 ; i< tab.length()-1 ; i++){
            text.setText(text.getText()+ tab.charAt(i));
        }
    }

    if(e.getSource() == addition){
        num1 = Double.parseDouble(text.getText());
        text.setText("");
        operateur = '+';
    }

    if(e.getSource() == soustraction){
        num1 = Double.parseDouble(text.getText());
        text.setText("");
        operateur = '-';
    }

    if(e.getSource() == multi){
        num1 = Double.parseDouble(text.getText());
        text.setText("");
        operateur = '*';
    }

    if(e.getSource() == division){
        num1 = Double.parseDouble(text.getText());
        text.setText("");
        operateur = '/';
    }

    if(e.getSource() == egal){
        num2 = Double.parseDouble(text.getText());
        text.setText("");
        switch (operateur){
            case '+':
                resultat = num1 + num2;
                break;
            case '*':
                resultat = num1 * num2;
                break;
            case '/':
                resultat = num1 / num2;
                break;
            case '-':
                resultat = num1 - num2;
                break;
        }
        text.setText(String.valueOf(resultat));
    }

    if(e.getSource() == point){
        text.setText(text.getText() + ".");
    }

    if (e.getSource() == negatif){
        Double achanger = Double.parseDouble(text.getText())*-1;
        text.setText(String.valueOf(achanger));
    }
}

```

## Conclusion

Cette calculatrice permet à l'utilisateur de réaliser des opérations mathématiques simples et d'utiliser des fonctionnalités supplémentaires comme l'inversion du signe

ou l'ajout d'un point décimal. Toutefois, des améliorations sont possibles pour rendre le programme plus robuste et plus fluide.