



# Práctica Calculadora

La siguiente práctica se refiere a una calculadora sencilla con 2 operandos para ello se deben realizar los siguientes campos:

1. Crear un nuevo proyecto en NetBeans: Graficos\_Calculadora.
2. Dentro de Source Packages: agregar una nueva clase: JFrame form: Calculadora.
3. Poner en null el layout del frame para mover los componentes libremente, con botón derecho en la propiedad Set-Layout y luego Null- Layout.
4. Agregar un JPanel.
5. Poner en null el layout del JPanel, con botón derecho en la propiedad Set-Layout y luego Null- Layout.
6. Modificar el `maximumSize` = [500, 400], `width`=500, `height`= 400.
7. A nivel de clase y antes del constructor, declarar las siguientes variables:  
`double operando1, operando2, resultado;`  
`char operador;`
8. Dentro del constructor de la clase y después del método  `initComponents()`, declarar la localización del frame:  
`this.setLocationRelativeTo(null);`
9. Agregar los siguientes componentes como se muestra en la imagen.
10. Agregar un `JTextField` y en las propiedades: `text=""`, `Font=Tahoma plain 36`, `HorizontalAlignment= right`, `editable= quitarle selección`. Cambiar el nombre al componente como: pantalla
11. Agregar los botones del 1 al 9 y renombrarlos como  `boton1`, ....  `boton9`.

Con tamaño de `width= 80`, `height=50`.



12. Agregar un botón para el cero Con tamaño de width= 180, heigth=50, cambiar nombre como: boton0
13. Agregar un botón para el punto ( . ) Con tamaño de width= 80, heigth=50, cambiar nombre como: botonpunto
14. Agregar los botones ce,c,+,-,\*,/ cuyo tamaño deben ser : width= 65, heigth=50
15. 

	Cambiar	nombre:
botonce,botonc,botonmas,botonmenos,botonpor,botondiv,botonigua		
16. El fondo para los botones: tahoma bold 14
17. Agregar en cada evento performed de cada botón la siguiente instrucción según corresponda con los botones del 0 al 9, por ejemplo: si se oprime el 9:  
`pantalla.setText(pantalla.getText()+"9");`
18. Para el botón del punto es necesario revisar que en la cadena de caracteres no haya un punto previo para no duplicarlo, entonces en el evento, performed del botón punto.  
Poner el siguiente código:

```
        boolean hayPunto= false;
        for(int i=0; i<pantalla.getText().length();i++){
            if(pantalla.getText().charAt(i)=='.'){
                hayPunto=true;
                break;}
        }
        if(!hayPunto){
            pantalla.setText(pantalla.getText()+'.');
        }
```
19. Para el botón CE, que corresponde a quitar el último carácter escrito es necesario que en el action performed del botón CE, poner el siguiente código:

```
        if(pantalla.getText().length()>0){
            pantalla.setText(pantalla.getText().substring(0,pantalla.getText().length()-1));
        }
```
20. Para el botón C que limpia el cuadro de texto es necesario poner el siguiente código en el evento performed:  
`pantalla.setText("");`



21. Para los botones +, \*, /, -, poner el siguiente código según corresponda en el evento action performed:

```
if(pantalla.getText().length()>0){  
    operando1=Double.parseDouble(pantalla.getText());  
    operador='+';  
    pantalla.setText("");  
}
```

Recuerda que tiene cambiar el operador, según corresponda.

22. Para el botón igual es necesario que tome el último texto y asignarlo al operador2, de acuerdo al siguiente código en el evento action performed:

```
if(pantalla.getText().length()>0){  
    operando2=Double.parseDouble(pantalla.getText());  
    switch(operador){  
        case '+':  
            resultado= operando1+operando2;  
            pantalla.setText(Double.toString(resultado));  
            break;  
        case '-':  
            resultado= operando1-operando2;  
            pantalla.setText(Double.toString(resultado));  
            break;  
        case '*':  
            resultado= operando1*operando2;  
            pantalla.setText(Double.toString(resultado));  
            break;  
        case '/':  
            resultado= operando1/operando2;  
            pantalla.setText(Double.toString(resultado));  
            break;  
    }
```

23. Probarla.