

## GUI : Graphical User Interface

### I. Định nghĩa và phân loại

Giao diện người dùng đồ họa (GUI) là giao diện kỹ thuật số trong đó người dùng tương tác với các thành phần đồ họa như biểu tượng, nút và menu, form... Trong GUI, hình ảnh được hiển thị trong giao diện có nhiệm vụ truyền tải thông tin liên quan đến người dùng, cũng như các hành động mà họ có thể thực hiện

Trong Java có các thư viện hỗ trợ xây dựng giao diện người dùng : AWT, Swing, JavaFX

-AWT (Abstract Window Toolkit): Khi Java mới ra đời thì AWT là một thư viện chứa các lớp giao diện, AWT có thể được sử dụng để xây dựng những giao diện đồ họa người dùng đơn giản nhưng không phù hợp với các dự án GUI yêu cầu tính toàn diện và phức tạp. Hơn nữa AWT phụ thuộc nền tảng nên dễ gặp lỗi

-Swing : Là một thư viện mạnh mẽ, đầy đủ, linh hoạt hơn so với AWT, ít phụ thuộc nền tảng như AWT. Swing được sử dụng để xây dựng các ứng dụng desktop

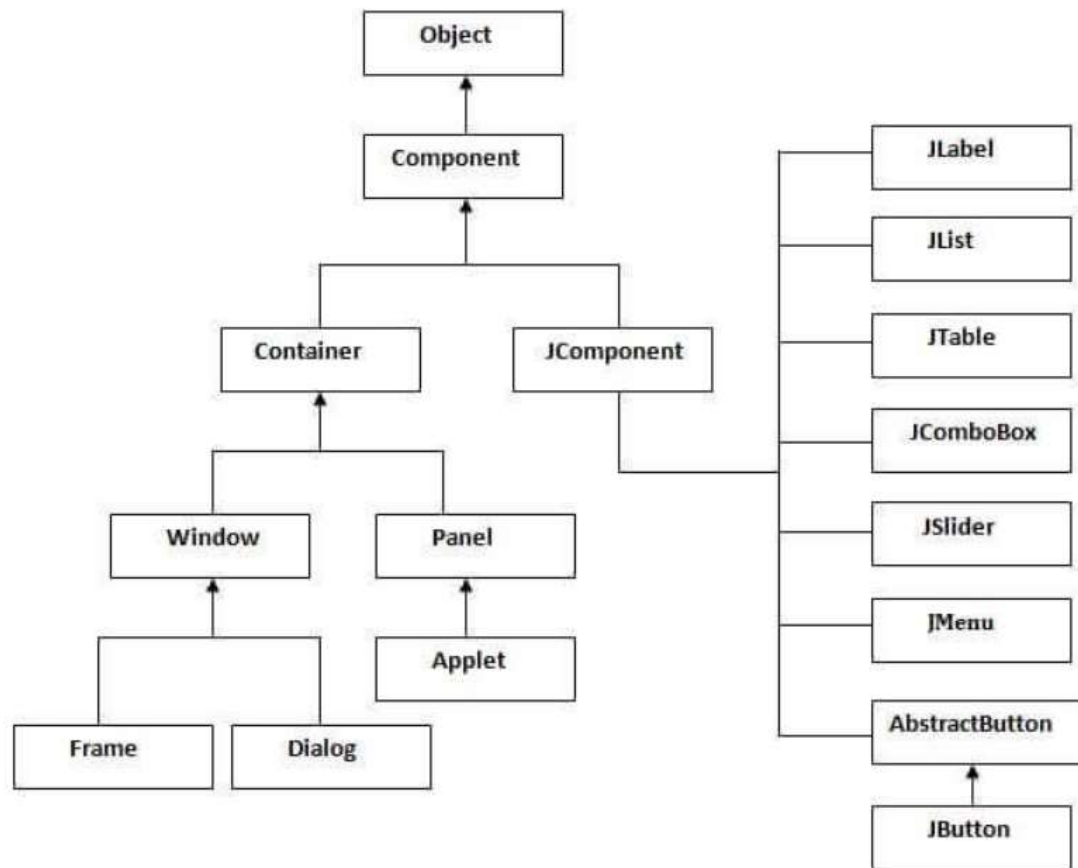
-JavaFX : Là thư viện đầy đủ và mạnh mẽ nhất, có thể sử dụng nhiều tính năng vượt trội hơn so với AWT và Swing. Ứng dụng JavaFX có thể chạy trên desktop, trình duyệt, tablet, smartphone...

Trong khóa học này, thư viện được sử dụng để học tập phần giao diện đồ họa người dùng là Swing

### II. Các thành phần chính của Java Swing

(Vẽ lại cho a cái hình này nhé)

Các lớp trong Java swing :



### 1.JFrame

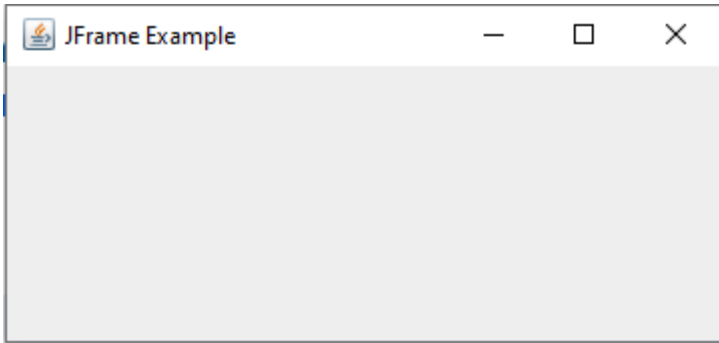
JFrame là một lớp container trong Java swing, nó như một cửa sổ chính chứa các thành phần khác như : textfield, button, labels...

Ví dụ 1:

```
import javax.swing.JFrame;

public class JFrameExample {
    public static void main(String[] args) {
        //create the frame
        JFrame frame = new JFrame("JFrame Example");
        frame.setSize(300, 100);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setLocationRelativeTo(null);
        frame.setVisible(true);
    }
}
```

Output :



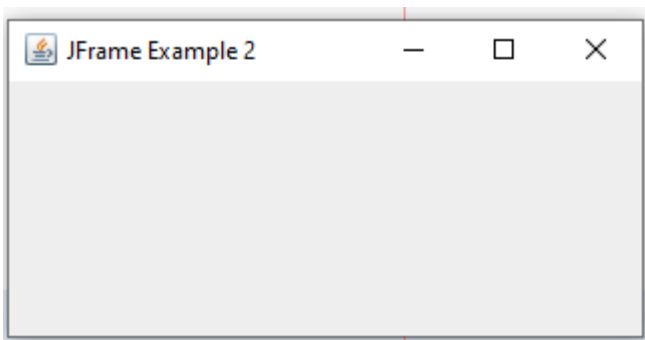
Ví dụ 2 :

Xây dựng lớp kế thừa từ lớp JFrame

```
import javax.swing.JFrame;

public class JFrameExample2 extends JFrame{
    public JFrameExample2(String title){
        this.setTitle(title);
        this.setSize(300, 100);
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        this.setLocationRelativeTo(null);
        this.setVisible(true);
    }
    public static void main(String[] args) {
        new JFrameExample2("JFrame Example 2");
    }
}
```

Output :



## 2.JButton

JButton hỗ trợ tạo ra một nút ấn có gắn nhãn và thực hiện một số sự kiện khi ta nhấn vào nó.

JButton được kế thừa từ lớp AbstractButton

Các hàm thông dụng của JButton :

Hàm	Chức năng
JButton()	Constructor
JButton(String s)	Constructor tạo button với nhãn s
setText(String s)	Gán nhãn cho button
getText()	Trả về nhãn trên button
void addActionListener(ActionListener v)	Thêm xử lý sự kiện cho button

Ví dụ 1 :

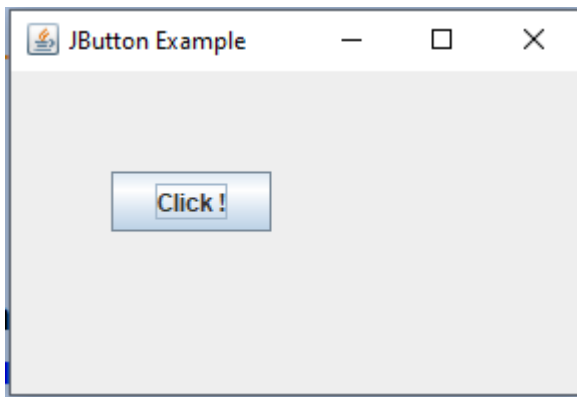
```
import javax.swing.JButton;
import javax.swing.JFrame;

public class JButtonExample {
    public static void main(String[] args) {
        JFrame frame = new JFrame("JButton Example");

        JButton btn = new JButton("Click !");
        btn.setBounds(50, 50, 80, 30);

        frame.add(btn);
        frame.setSize(300, 200);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setLocationRelativeTo(null);
        frame.setLayout(null);
        frame.setVisible(true);
    }
}
```

Output :



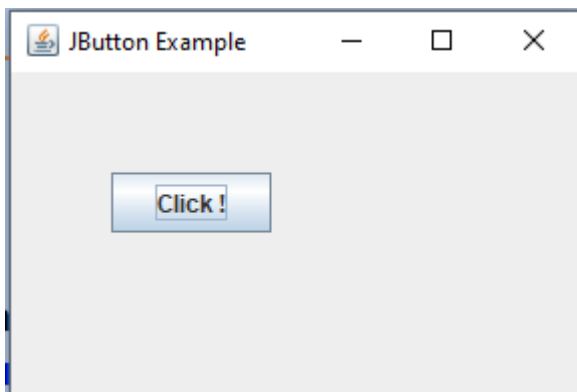
Ví dụ 2 :

```
public class JFrameExample2 extends JFrame{
    private JButton btn;

    public JFrameExample2(String title){
        this.setTitle(title);
        btn = new JButton("Click !");
        btn.setBounds(100, 50, 80, 30);
        this.setSize(300, 200);
        this.add(btn);

        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        this.setLocationRelativeTo(null);
        this.setLayout(null);
        this.setVisible(true);
    }
    public static void main(String[] args) {
        new JFrameExample2("JFrame Example 2");
    }
}
```

Output :



### Ví dụ 3 : JButton vs ActionListener interface

Sử dụng phương thức addActionListener() cho button, khi button được click thì phương thức actionPerformed() sẽ được thực thi.

```
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JTextField;

public class JButtonExample {
    public static void main(String[] args) {
        JFrame frame = new JFrame("JButton Example");

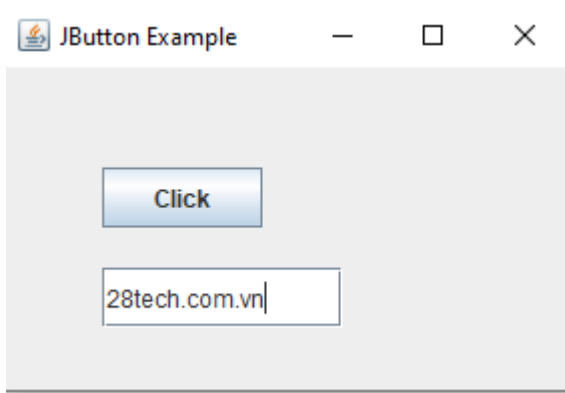
        JButton btn = new JButton("Click");
        btn.setBounds(50, 50, 80, 30);

        JTextField jtf = new JTextField();
        jtf.setBounds(50, 100, 120, 30);

        btn.addActionListener(new ActionListener(){
            @Override
            public void actionPerformed(ActionEvent e) {
                jtf.setText("28tech.com.vn");
            }
        });
        frame.add(jtf);
        frame.add(btn);

        frame.setSize(300, 200);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setLocationRelativeTo(null);
        frame.setLayout(null);
        frame.setVisible(true);
    }
}
```

Output :



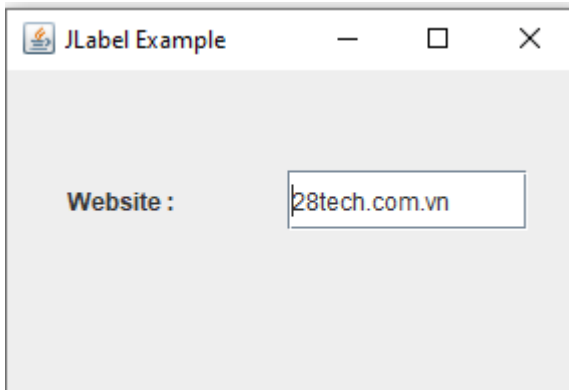
### 3.JLabel

JLabel được sử dụng làm nhãn, hiển thị text, icon trên giao diện đồ họa người dùng.

Ví dụ :

```
public class Example {  
    public static void main(String[] args) {  
        JFrame frame = new JFrame("JButton Example");  
  
        JLabel label = new JLabel("Website : ");  
        label.setBounds(30, 50, 80, 30);  
        JTextField jtf = new JTextField("28tech.com.vn");  
        jtf.setBounds(140, 50, 120, 30);  
  
        frame.add(jtf);  
        frame.add(label);  
  
        frame.setSize(300, 200);  
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        frame.setLocationRelativeTo(null);  
        frame.setLayout(null);  
        frame.setVisible(true);  
    }  
}
```

Output :



#### 4.JTextField

JTextField có nhiệm vụ hiển thị, chứa, chỉnh sửa nội dung text trên 1 dòng.

Ví dụ 1:

```
public class Example {
    public static void main(String[] args) {
        JFrame frame = new JFrame("JTextField Example");

        JLabel label = new JLabel("Website : ");
        label.setBounds(30, 50, 80, 30);
        JTextField jtf = new JTextField("28tech.com.vn");
        jtf.setBounds(140, 50, 120, 30);

        frame.add(jtf);
        frame.add(label);

        frame.setSize(300, 200);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setLocationRelativeTo(null);
        frame.setLayout(null);
        frame.setVisible(true);
    }
}
```

Ví dụ 2 :

```
public class JTextFieldExample implements ActionListener{
    private JButton btn1, btn2;
    private JTextField link;
    private JFrame frame;
    JTextFieldExample(){
```



```

    btn1 = new JButton("Website");
    btn2 = new JButton("Youtube");
    link = new JTextField("");
    frame = new JFrame("JTextField Example");
    btn1.setBounds(50, 50, 100, 30);
    btn2.setBounds(150, 50, 100, 30);
    link.setBounds(50, 100, 180, 30);
    frame.add(btn1);
    frame.add(btn2);
    frame.add(link);

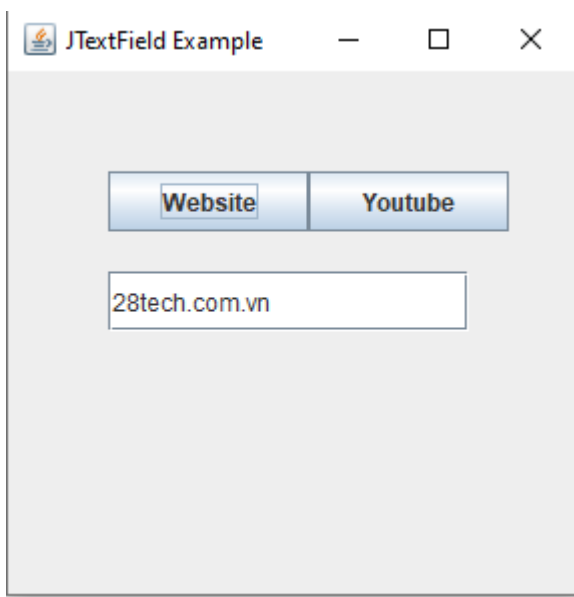
    btn1.addActionListener(this);
    btn2.addActionListener(this);

    frame.setSize(300, 300);
    frame.setLocationRelativeTo(null);
    frame.setLayout(null);
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    frame.setVisible(true);
}
@Override
public void actionPerformed(ActionEvent e) {
    if(e.getSource() == btn1){
        link.setText("28tech.com.vn");
    }
    else{
        link.setText("youtube.com/28tech_");
    }
}

public static void main(String[] args) {
    new JTextFieldExample();
}
}

```

Output :



### 5. JTextArea

JTextArea giúp hiển thị, chỉnh sửa text trên nhiều dòng.

Các hàm phổ biến của JTextArea :

Hàm	Chức năng
JTextArea(String s)	Constructor tạo JTextArea với nội dung là s
JTextArea(int row, int column)	Constructor tạo JTextArea với số hàng, cột
getText()	Trả về nội dung trong text area
setText()	Gán nội dung cho text area
setLineWrap(boolean)	Tự động xuống dòng khi nội dung bị tràn ở dòng hiện tại nếu tham số là true
setWrapStyleWord(boolean)	Tự động gói từ nếu tham số là false

Ví dụ 1:

```
public class TextAreaExample {

    public static void main(String[] args) {
        JFrame frame = new JFrame("JTextArea Example");

        JLabel label = new JLabel("Content : ");
        label.setBounds(30, 50, 80, 30);
        JTextArea jta = new JTextArea();
        jta.setBounds(140, 50, 200, 200);
```

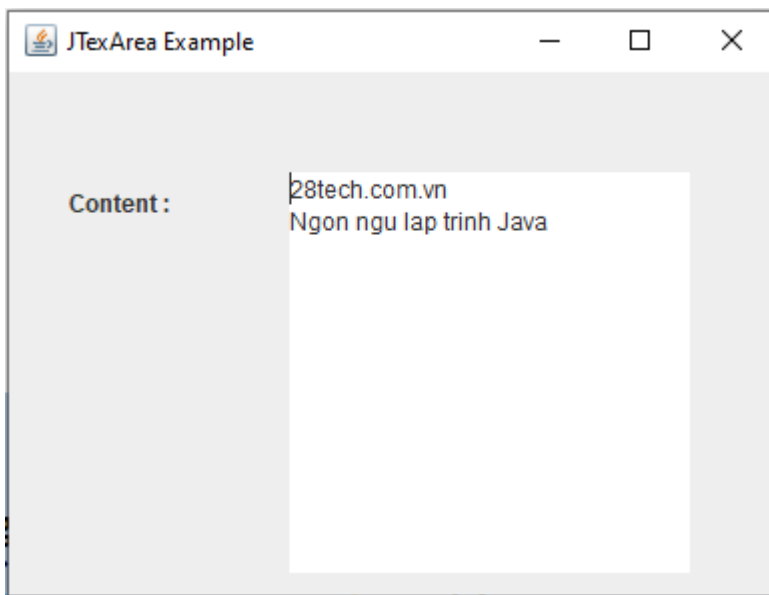
```

jta.setText("28tech.com.vn\nNgon ngu lap trinh Java");
jta.setWrapStyleWord(true);
jta.setLineWrap(true);
frame.add(jta);
frame.add(label);

frame.setSize(400, 300);
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
frame.setLocationRelativeTo(null);
frame.setLayout(null);
frame.setVisible(true);
}
}

```

Output :



Ví dụ 2 : Đếm từ

```

public class TextAreaExample {

    public static void main(String[] args) {
        JFrame frame = new JFrame("JTextArea Example");

        JLabel label = new JLabel("Data : ");
        label.setBounds(30, 30, 80, 30);
        JTextArea jta = new JTextArea();
        jta.setBounds(140, 30, 200, 200);
    }
}

```

```

        JButton btn = new JButton("Count");
        btn.setBounds(140, 250, 80, 30);
        JTextField jtf = new JTextField("");
        jtf.setBounds(240, 250, 80, 30);

        jta.setText("28tech.com.vn\nNgon ngu lap trinh Java");
        jta.setWrapStyleWord(true);
        jta.setLineWrap(true);

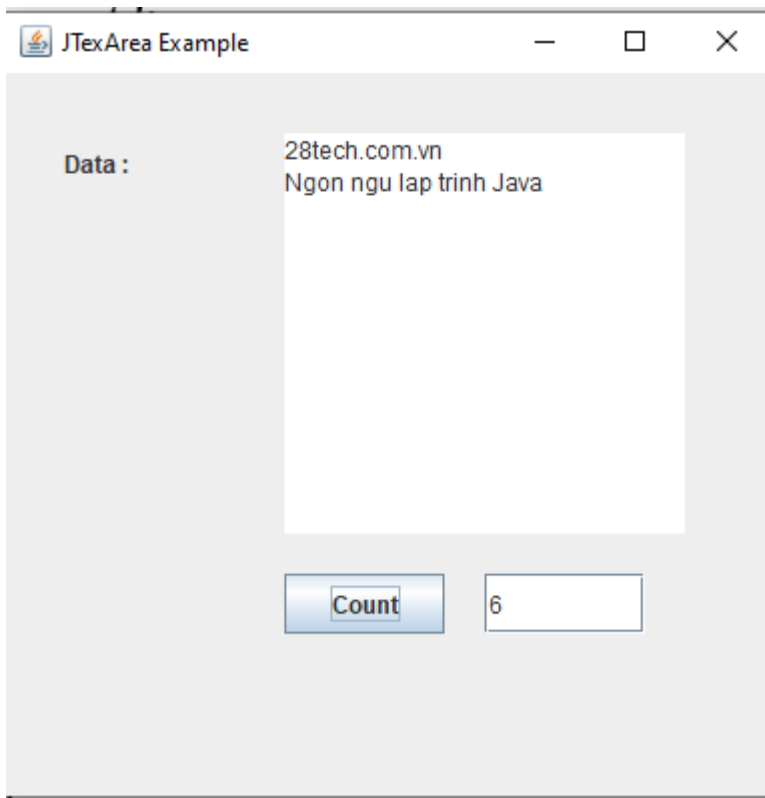
        btn.addActionListener(new ActionListener(){
            @Override
            public void actionPerformed(ActionEvent e) {
                String s = jta.getText();
                String[] word = s.split("\\s+");
                jtf.setText(word.length + "");
            }
        });

        frame.add(jta);
        frame.add(label);
        frame.add(btn);
        frame.add(jtf);

        frame.setSize(400, 400);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setLocationRelativeTo(null);
        frame.setLayout(null);
        frame.setVisible(true);
    }
}

```

Output :



#### 6.JPasswordField

JPasswordField được sử dụng để lưu trữ nội dung là mật khẩu

Ví dụ :

```
public class TextAreaExample {  
  
    public static void main(String[] args) {  
        JFrame frame = new JFrame("JPasswordField Example");  
        JLabel lbl1 = new JLabel("Username : ");  
        JLabel lbl2 = new JLabel("Password : ");  
  
        lbl1.setBounds(50, 50, 100, 30);  
        lbl2.setBounds(50, 100, 100, 30);  
  
        JTextField username = new JTextField();  
        JPasswordField password = new JPasswordField();  
  
        username.setBounds(150, 50, 150, 30);  
        password.setBounds(150, 100, 150, 30);  
  
        JButton login = new JButton("Login");  
        login.setBounds(100, 150, 80, 30);  
    }  
}
```

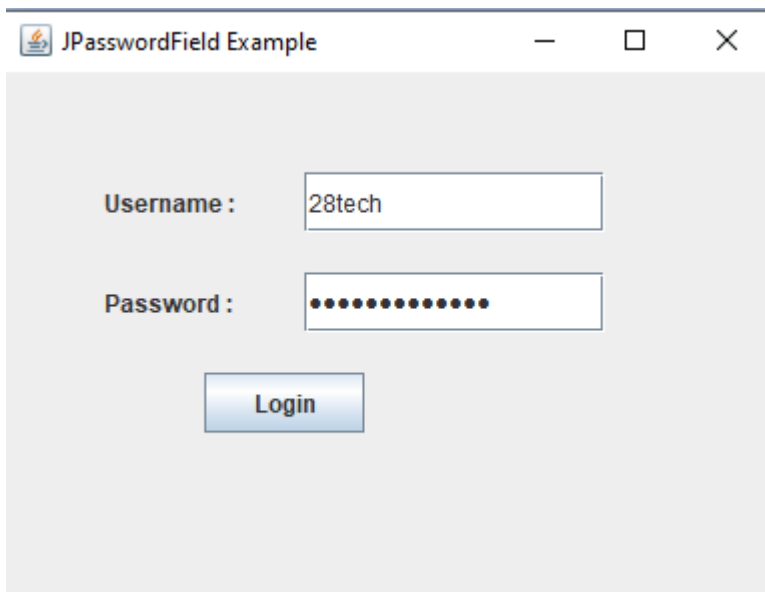
```

        frame.add(lbl1);
        frame.add(lbl2);
        frame.add(username);
        frame.add(password);
        frame.add(login);

        frame.setSize(400, 300);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setLocationRelativeTo(null);
        frame.setLayout(null);
        frame.setVisible(true);
    }
}

```

Output :



## 7. JOptionPane

JOptionPane được sử dụng để hiển thị các hộp thông báo, hộp xác nhận, hộp đầu vào..

```

public class TextAreaExample {

    public static void main(String[] args) {
        JFrame frame = new JFrame("JPasswordField Example");
        JLabel lbl1 = new JLabel("Username : ");
        JLabel lbl2 = new JLabel("Password : ");
    }
}

```

```

lbl1.setBounds(50, 50, 100, 30);
lbl2.setBounds(50, 100, 100, 30);

JTextField username = new JTextField();
JPasswordField password = new JPasswordField();

username.setBounds(150, 50, 150, 30);
password.setBounds(150, 100, 150, 30);

JButton login = new JButton("Login");
login.setBounds(100, 150, 80, 30);

frame.add(lbl1);
frame.add(lbl2);
frame.add(username);
frame.add(password);
frame.add(login);

login.addActionListener(new ActionListener(){
    @Override
    public void actionPerformed(ActionEvent e) {
        if(username.getText().equals("28tech") &&
String.valueOf(password.getPassword()).equals("28tech")){
            JOptionPane.showMessageDialog(frame, "Successful !");
        }
        else{
            JOptionPane.showMessageDialog(frame, "Failed !");
        }
    }
});
frame.setSize(400, 300);
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
frame.setLocationRelativeTo(null);
frame.setLayout(null);
frame.setVisible(true);
}
}

```

Output :

