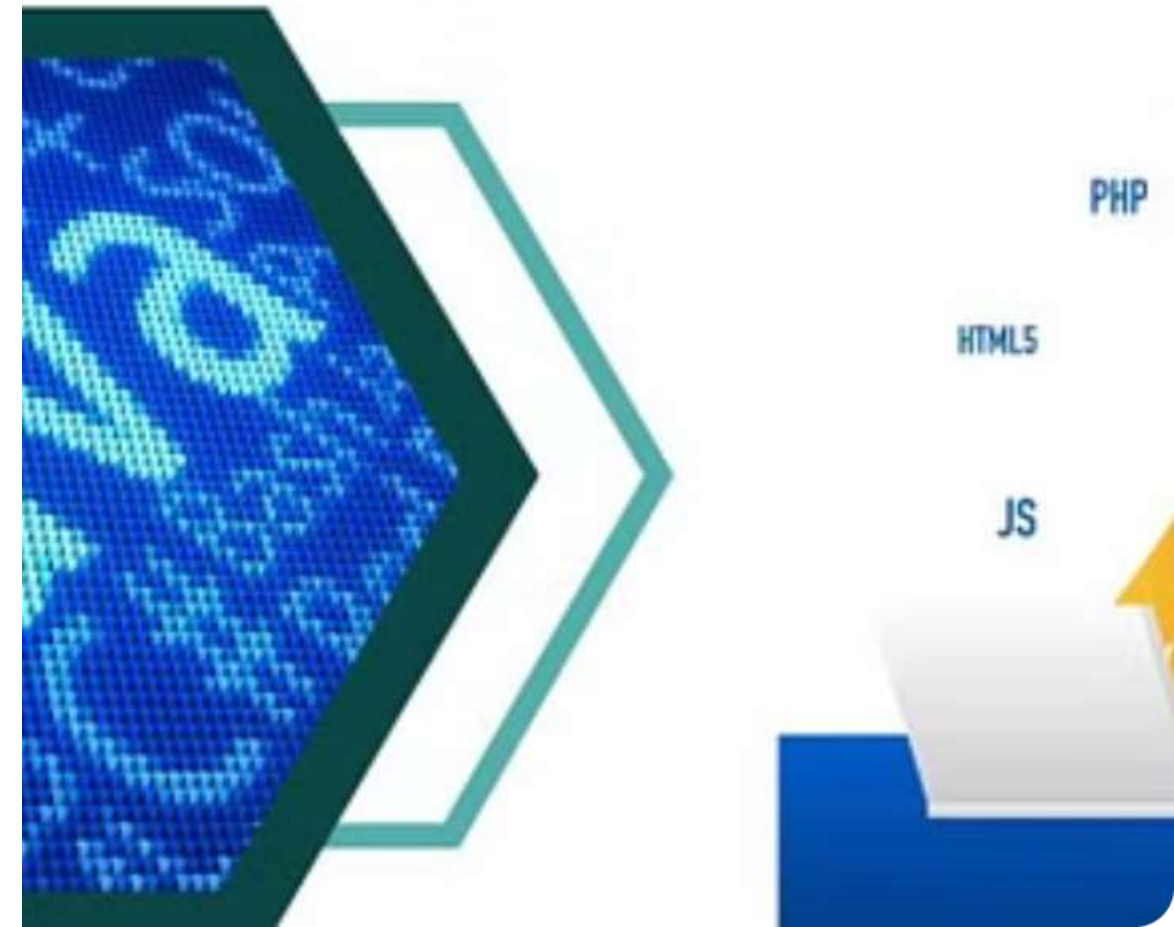


Introduction to Swing Components

Swing is a set of GUI widgets and components that allows Java developers to create desktop applications with visually appealing and interactive user interfaces. These components provide a wide range of functionality, from simple buttons to complex text editors.

Swing Components Java





JButton

1 Clickable

JButton is a component that the user can click to perform an action, such as triggering an event or opening a new window.

2 Customizable

JButton can be customized with text, icons, or both, and its appearance can be modified to match the application's design.

3 Versatile

JButton can be used for a variety of purposes, from simple actions to complex functionality within a user interface.

JLabel

Text Display

JLabel is a component that displays static text, which can be used to provide labels, headings, or instructions to the user.

Image Display

JLabel can also be used to display images, allowing for the integration of visual elements within the user interface.

Styling Options

The appearance of JLabel can be customized, including the font, color, and alignment, to ensure it fits the overall design of the application.

JTextField

Text Input

JTextField is a component that allows the user to enter text, which can be used for various purposes, such as login forms or search bars.

Validation

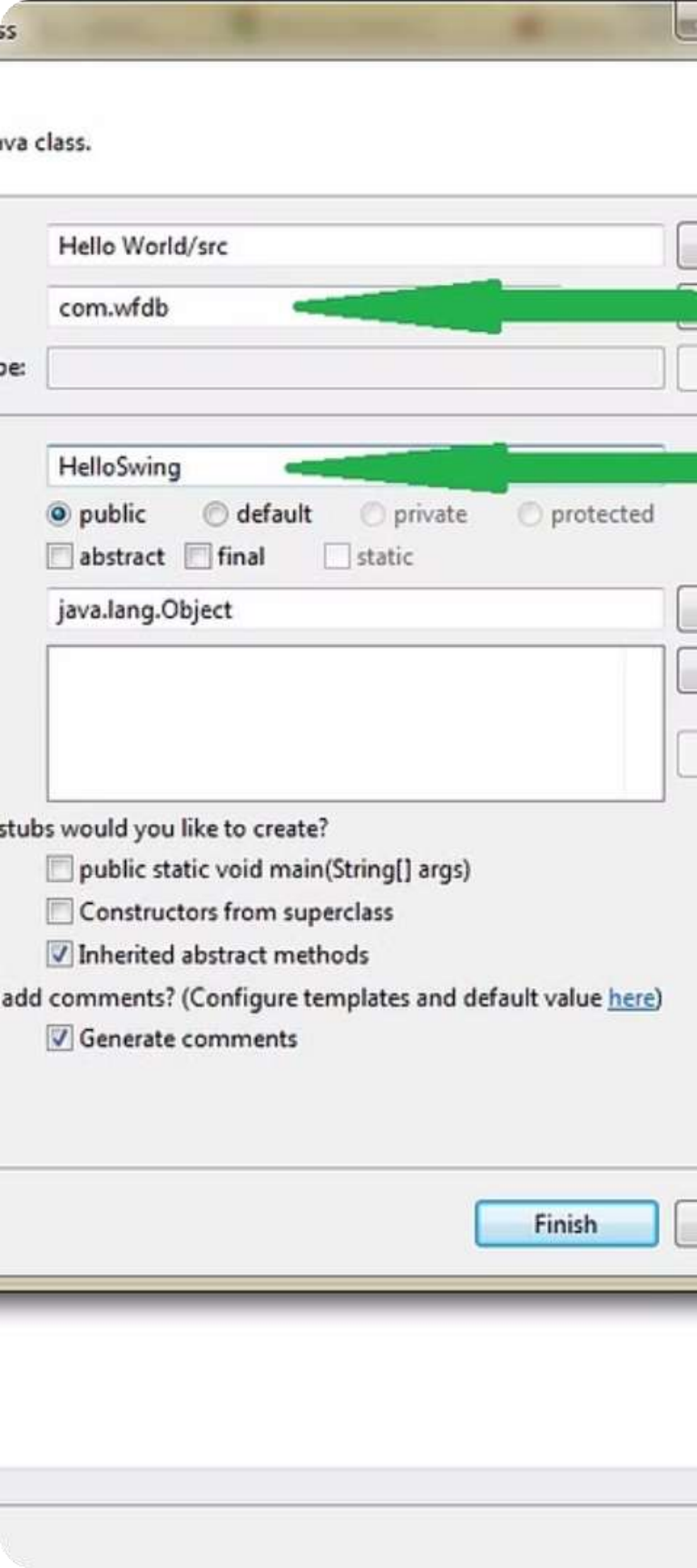
JTextField can be configured to perform input validation, ensuring that the user enters data in the expected format.

Event Handling

Developers can add event listeners to JTextField to respond to user actions, such as text changes or focus events.

Customization

The appearance and behavior of JTextField can be customized to match the design and functionality requirements of the application.

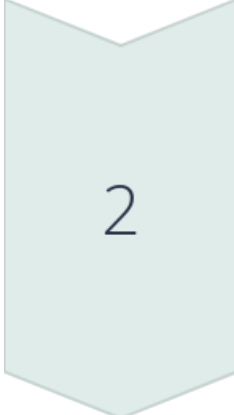


JTextArea



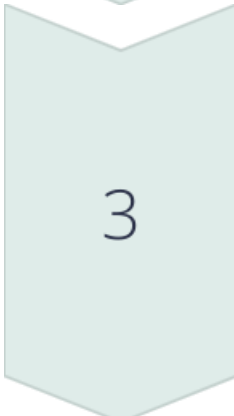
Multi-Line Input

JTextArea is a component that allows the user to enter multiple lines of text, making it suitable for tasks such as writing notes or entering longer form content.



Scrolling

JTextArea can be configured to include vertical and/or horizontal scrollbars, enabling the user to access content that exceeds the visible area of the component.



Formatting Options

Developers can apply various formatting options to JTextArea, such as font styles, line spacing, and text alignment, to enhance the visual presentation of the content.

JCheckBox



Selection

JCheckBox is a component that allows the user to select or deselect an option, providing a simple and intuitive way to make choices.



Grouping

Multiple JCheckBoxes can be grouped together, enabling the user to select one or more options from a set of related choices.



Toggling

The state of a JCheckBox (selected or deselected) can be easily toggled by the user, providing a responsive and interactive user interface.

JRadioButton

1

Exclusive Selection

JRadioButton is a component that allows the user to select one option from a mutually exclusive group of choices.

2

Grouping

JRadioButtons must be grouped together using a ButtonGroup to ensure that only one option can be selected at a time.

3

Visual Feedback

The selected JRadioButton is visually highlighted, providing clear feedback to the user about the current state of the choice.

```
ing.*;
.event.*;
onExample extends JFrame implements Action
1,rb2;

ple(){
utton("Male");
00,50,100,30);
utton("Female");
00,100,100,30);
new ButtonGroup();
add(rb2);
click");
,150,80,30);
ener(this);
2);add(b);
);
);

ionPerformed(ActionEvent e){
ed()){
wMessageDialog(this,"You are Male.");

ed()){
wMessageDialog(this,"You are Female.");

oid main(String args[]){
Example();
```

JComboBox

Drop-Down Menu

JComboBox displays a list of options in a drop-down menu, allowing the user to select one item from the list.

Editable

JComboBox can be configured to be editable, enabling the user to type in a custom value or select from the predefined options.

Event Handling

Developers can add event listeners to JComboBox to respond to user actions, such as selecting an item or typing in a new value.

Customization

The appearance and behavior of JComboBox can be customized to match the overall design and functionality of the application.

JSlider

Range Selection

JSlider is a component that allows the user to select a value from a continuous range, making it suitable for tasks such as volume control or setting a preference.

Visualization

JSlider can be customized to display tick marks, labels, and a thumb (the movable component) to provide visual feedback to the user.

Event Handling

Developers can add event listeners to JSlider to respond to user actions, such as dragging the thumb or releasing it, allowing for dynamic updates to the application's state.