

Question: What are the differences between Swing and AWT implementation?

1, Top-Level Containers:

-AWT: + The primary top-level container used in AWT is Frame. It represents a standard window with title, border, and buttons.

+ AWT top-level containers are heavyweight and depend on the native windowing system.

-Swing: + The primary top-level containers used in AWT are JFrame, JWindow, JDialog and JApplet. Swing top-level containers are lightweight and more flexible.

+ They provide pluggable look-and-feel (PLAF) and more control over component rendering.

2, Component Class Names:

-AWT: Component names are often generic, like Button, TextField, and Checkbox.

-Swing: Swing components have a more descriptive naming convention, like JButton, JTextField, and JCheckBox. This improves the code readability and maintainability.

3, Event Handling:

-AWT: Relies on a single listener interface (EventListener) for all events. This can lead to cluttered code.

-Swing: Uses a more specific event model with dedicated listener interfaces for different events

For example, ActionListener for button clicks, MouseListener for mouse interactions.

=>This improves code organization and clarity.

4, Appearance:

-AWT: The appearance of AWT components is determined by the native GUI toolkit of the platform they run on.

=>This means that AWT applications look different on different operating systems.

-Swing: Swing supports pluggable look-and-feel, which means you can change the appearance of your application to mimic different operating systems