

Графични интерфейси с Java

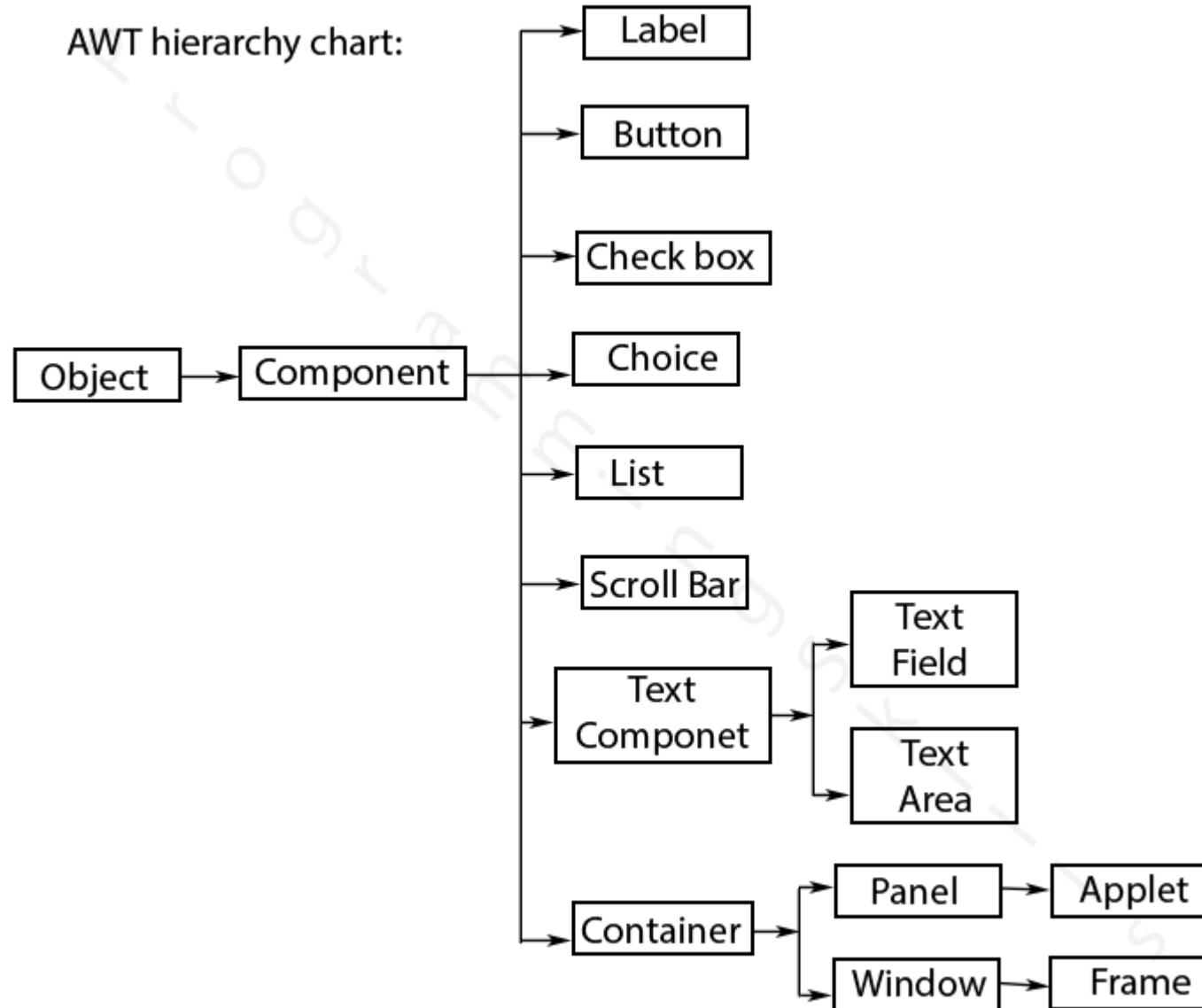
Лекции 8.1

ОСНОВНИ ПОНЯТИЯ

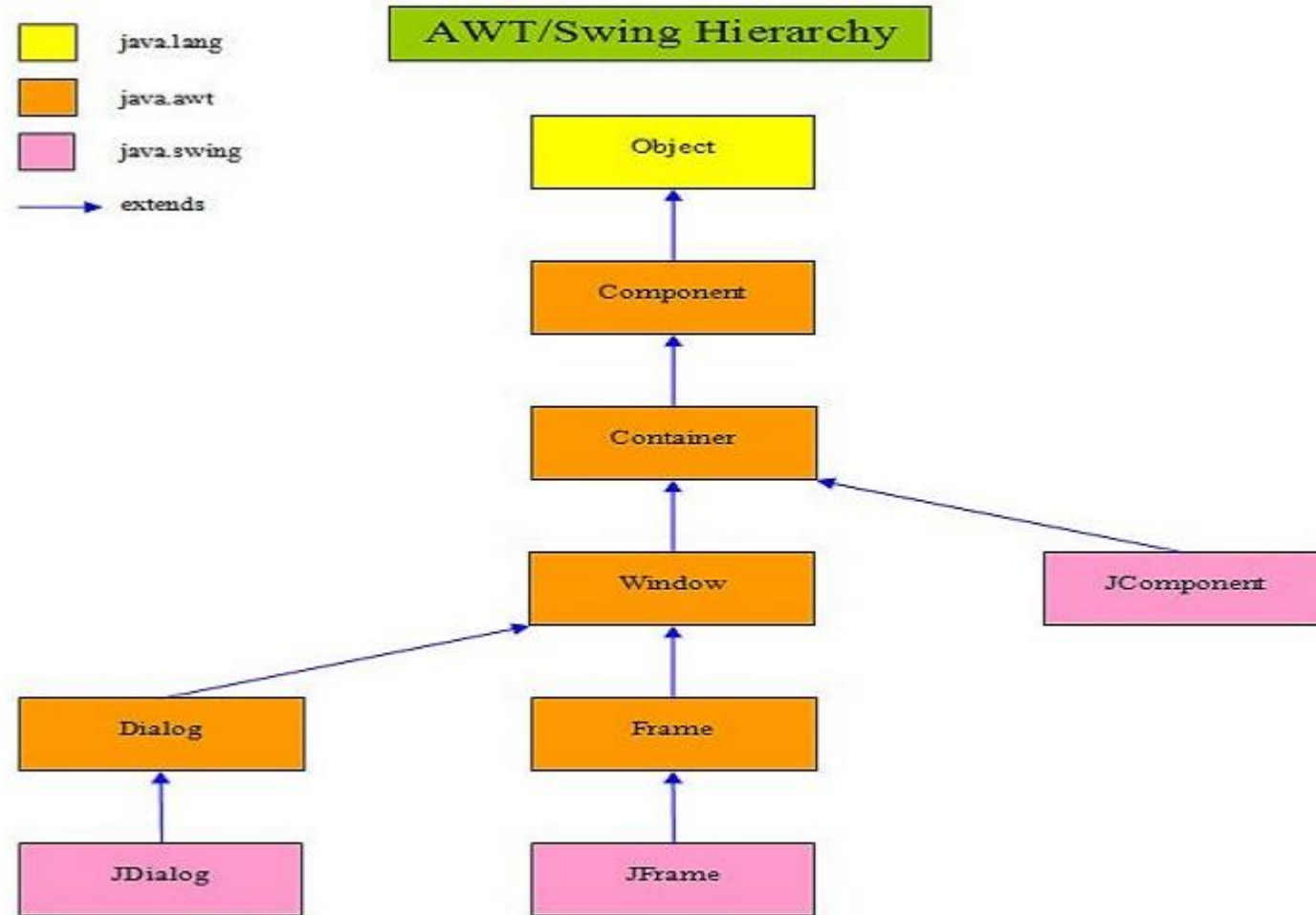
Разликата между конзолна програма и програма с графичен интерфейс е в управлението на събитията.

- В първия случай, програмата управлява потребителя изисквайки от него доставянето на информация в определени моменти. Диалогът се води в текстов прозорец наречен конзола.
- Във втория случай потребителят управлява програмата, която реагира на негови команди, които той отправя чрез събития върху компоненти (прозорци, бутони, текстови компоненти ...)
- **компонент** – базов обект от потребителския интерфейс; всичко, което се изобразява на екрана: **прозорци, зони за рисуване, бутони, полета за отметки, текстови полета, скролиращи ленти ...**
- **контейнер** – обединява група компоненти, предоставя ги за изобразяване и ги свързва с някаква периферия.

Abstract Windowing Toolkit



Swing



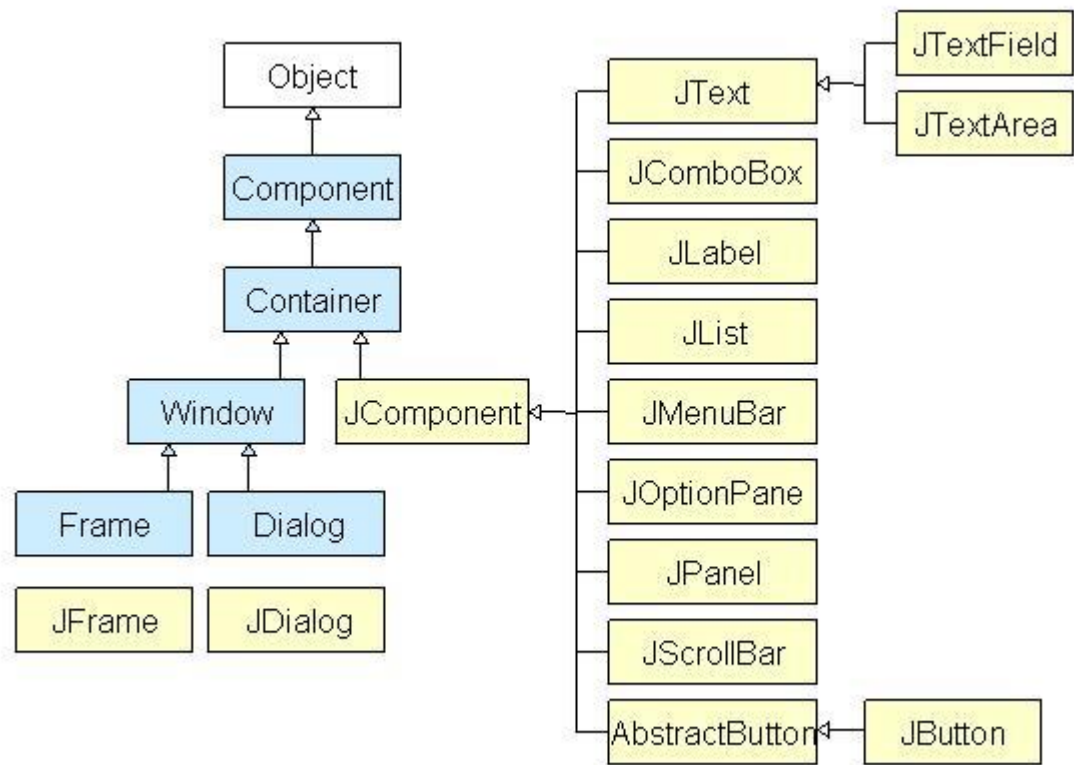
Swing - 1

Java AWT	Java Swing
AWT components are platform-dependent.	Java swing components are platform-independent.
AWT components are heavyweight.	Swing components are lightweight.
AWT doesn't support pluggable look and feel.	Swing supports pluggable look and feel.
AWT provides less components than Swing.	Swing provides more powerful components such as tables, lists, scrollpanes, colorchooser, tabbedpane, etc.
AWT doesn't follow MVC(Model View Controller) where model represents data, view represents presentation and controller acts as an interface between model and view.	Swing follows MVC.

Swing -2

<i>Component</i>	AWT
<i>Label</i>	AWT
<i>Container</i>	AWT
JComponent	
JLabel	
JComboBox	
AbstractButton	
JButton	
JToggleButton	
JCheckBox	
JRadioButton	
JMenuItem	
JMenu	
JCheckBoxMenuItem	
JMenuBar	
JPanel	
JToolBar	
JScrollPane	
JTabbedPane	
JSplitPane	
JTextComponent	
JTextField	
JTextArea	
JEditorPane	
JList	
JTree	
JTable	
<i>Window</i>	AWT
<i>Frame</i>	AWT
JFrame	
<i>Dialog</i>	AWT
JDialog	
Jwindow	

Йерархия на класовете в Swing



JFrame

```
import javax.swing.* ;  
public class Window1 {  
    public static void main (String args[]){  
        JFrame wd = new JFrame() ;  
        wd.setSize (250, 100) ;  
        wd.setTitle ("graphic window") ;  
        wd.setVisible (true) ;  
    }  
}
```



```
wd.setDefaultCloseOperation(JFrame.DO_NOTHING_ON_CLOSE);  
wd.setDefaultCloseOperation(JFrame.HIDE_ON_CLOSE);           // default)  
wd.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);  
wd.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```


Пример

```
import java.awt.event.*;
import javax.swing.* ;
public class Window1 {
    public static void main (String args[]){
        JFrame wd = new JFrame() ;
        wd.setSize (250, 100) ;
        wd.setTitle ("graphic window") ;
        wd.setVisible (true) ;
        wd.addWindowListener(new WndCls());
    }
    // To close the graphic thread:
    static class WndCls extends WindowAdapter {
        public void windowClosing(WindowEvent e) {
            System.exit(0);
        }
    }
}
```

```
import javax.swing.* ;
public class Window1 {
    public static void main (String args[]){
        JFrame wd = new JFrame() ;
        wd.setSize (250, 100) ;
        wd.setTitle ("graphic window") ;
        wd.setVisible (true) ;

        wd.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
}
```

Някои събития и съответните им слушатели

За всяко действие върху компонент в резултат, на което се получава събитие Xxx, се асоциира обект - слушател (от тип XxxEvent), чрез метод `addXxxListener`. Всеки път, когато слушателят притежава няколко метода трябва:

- или да се предифинират всички методи на съответния интерфейс XxxListener (който трябва да бъде наследен с `implements`), като някои от тях могат да не вършат нищо;
- или да се наследи съответният адаптер и да се предифинират само интересующите ни методи.

ОСНОВНИ ПОНЯТИЯ

Едно събитие може да бъде слушано от много слушатели, които се изпълняват един след друг.

Някои събития и съответните им слушатели	
Действия в резултат, на които се получава събитие	тип на слушателя
"clicks" на бутон, "Enter" в текстово поле или избор на "menu item"	ActionListener
затваряне на "frame" (main window)	WindowListener
"clicks" на бутон на мишката, когато "cursor" се намира върху компонент	MouseListener
преместване на мишката върху компонент	MouseMotionListener
компонентът става видим	ComponentListener
компонентът влиза във фокуса на клавиатурата	FocusListener
промяна в избора на даден "list"	ListSelectionListener
промяна на характеристиките(като текст или "label") на компонента	PropertyChangeListener