GLS UNIVERSITY

SEM – III

0301301 - CORE JAVA

Dr. Disha Shah
Prof. Vidhi Thakkar

Unit – 5
Applets in Java

Index

Applets

- Introduction
- Applet class
- Applet Life cycle
- Applet Structure
- Common methods
  - drawString()
  - setBackground()
  - setForeground()
  - showStatus()
- Applet using Graphic class
  - drawLine()
  - setColor() and setFont()
  - drawOval() and fillOval()
  - drawRect() and fillRect()
- Basic Controls in Japplet
  - Jlable
  - Jbutton
  - JtextArea
  - JtextField
  - JRadioButton
  - JcheckBox
  - JCombobox

Applet in Java

- Applet is a special type of program that is embedded in the webpage to generate the dynamic content.
- It runs inside the browser and works at client side.
- In other words, we can say that Applets are small Java applications that can be accessed on an Internet server, transported over Internet, and can be automatically installed and run as apart of a web document.
- After a user receives an applet, the applet can produce a graphical user interface.
- It has limited access to resources so that it can run complex computations without introducing the risk of viruses or breaching data integrity.

Applet in Java

- **Advantage of Applet**

  There are many advantages of applet. They are as follows:

  - It works at client side so less response time.
  - Secured

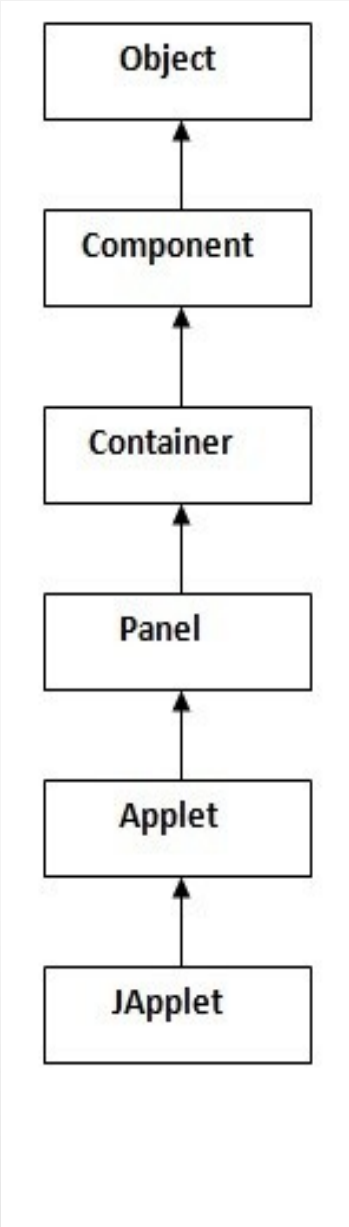
- It can be executed by browsers running under many platforms, including Linux, Windows, Mac Os etc.

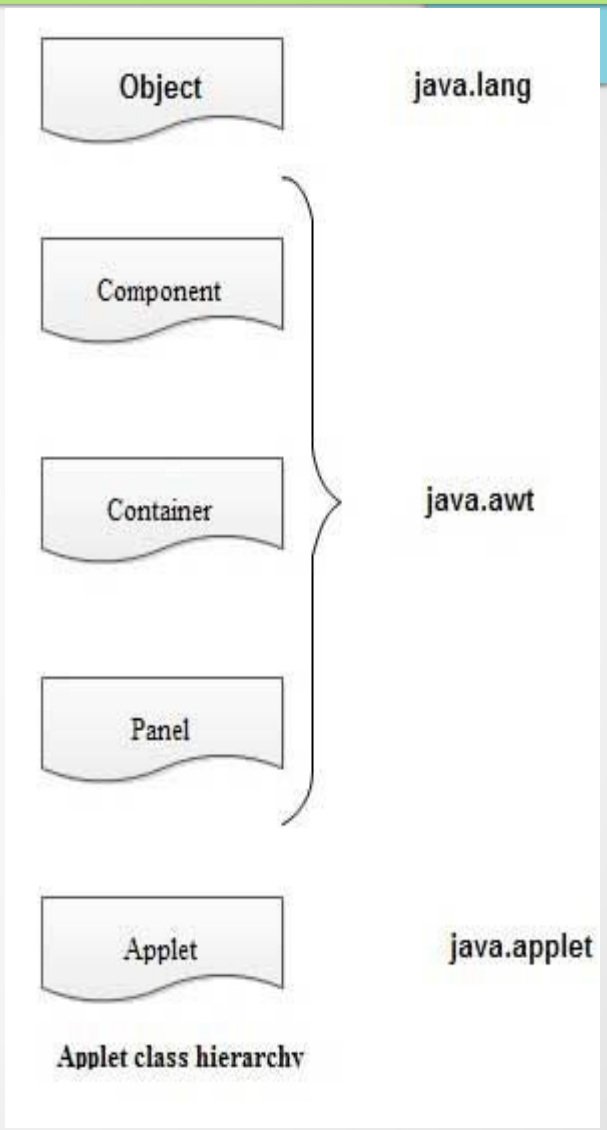
- **Drawback of Applet**
- Plugin is required at client browser to execute applet.

Difference between Applications & Applet

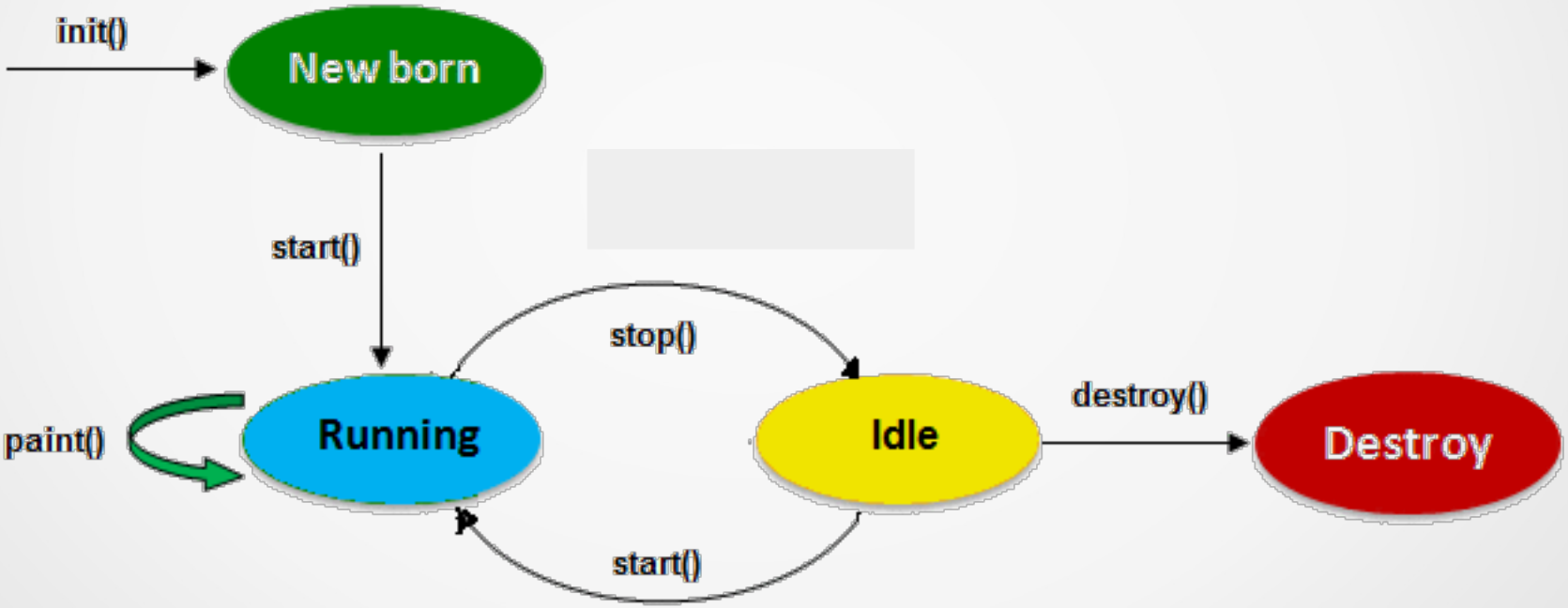
Applications	Applet
The execution of an application program starts from main().	The execution of the applet does not start from main() method.
These can run on their own. In order to get executed, they need not be embedded inside any web page.	Applets cannot run on their own. They have to be embedded inside a web page to get executed.
Applications are executed at command line.	Applets can be executed inside a web browser or appletviewer.
Applications have no inherent security restrictions.	Applets execute under strict security limitations that disallow certain operations.
Applications have their own life cycle.	Applets have their own life cycle. init -> start -> paint -> stop -> destroy

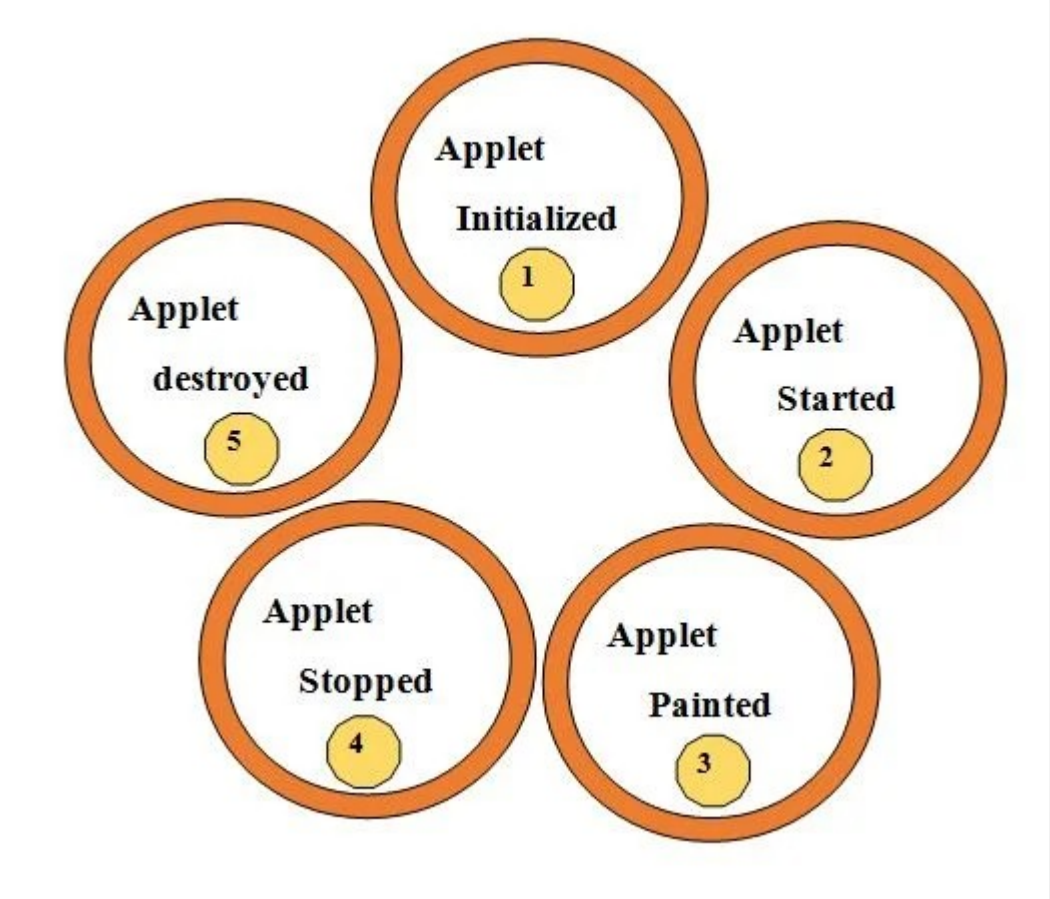
Hierarchy of Applet





Lifecycle of Java Applet




Applet
Initialized
1
Applet
Started
2
Applet
Painted
3
Applet
Stopped
4
Applet
destroyed
5

Lifecycle of Java Applet

- Applet is initialized.
- Applet is started.
- Applet is painted.
- Applet is stopped.
- Applet is destroyed.

Lifecycle methods for Applet

- The java.applet.Applet class has 4 life cycle methods and java.awt. Component class provides 1 life cycle methods for an applet.

java.applet.Applet class

- For creating any applet java.applet.Applet class must be inherited.
- It provides 4 life cycle methods of applet.
- **public void init()**: is used to initialized the Applet. It is invoked only once.
- **public void start()**: is invoked after the init() method or browser is maximized. It is used to start the Applet.
- **public void stop()**: is used to stop the Applet. It is invoked when Applet is stop or browser is minimized.
- **public void destroy()**: is used to destroy the Applet. It is invoked only once.

Lifecycle methods for Applet

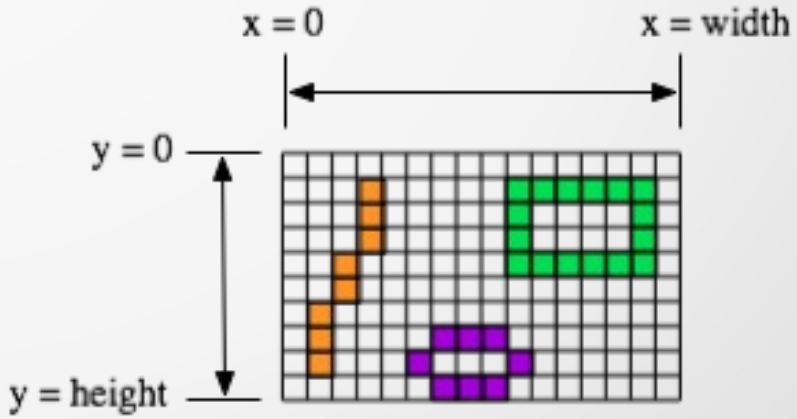
java.awt.Component class

- The Component class provides 1 life cycle method of applet.

- **public void paint(Graphics g)**: is used to paint the Applet.
- It provides Graphics class object that can be used for drawing oval, rectangle, arc etc.
- Java Plug-in software is responsible to manage the life cycle of an applet.

How to run an Applet?

- There are two ways to run an applet
  - By web browser
    - Compiling: javac Applet1.java
    - Create an Html file and embed the Applet tag in the HTML file.



How to run an Applet?

- By appletViewer tool
  - Write HTML APPLET tag in comments in the source file.
  - Compile the applet source code using javac.
  - Use applet viewer ClassName.class to view the applet.

    c:\>javac Applet1.java

    c:\>appletviewer Applet1.java

Displaying Graphics in Applet

- java.awt.Graphics class provides many methods for graphics programming.

  Commonly used methods of Graphics class:
- public abstract void drawString(String str, int x, int y)

  is used to draw the specified string.
- public void drawRect(int x, int y, int width, int height)

  draws a rectangle with the specified width and height.
- public void fillRect(int x, int y, int width, int height)

  is used to fill rectangle with the default color and specified width and height.
- public void drawOval(int x, int y, int width, int height)

  is used to draw oval with the specified width and height.
- public void fillOval(int x, int y, int width, int height)

  is used to fill oval with the default color and specified width and height.

Displaying Graphics in Applet

- public void drawRoundRect(int x,int y,int width,int height,int arcWidth,int arcHeight)

  is used to draw the rounded rectangle.

- public void fillRoundRect(int x, int y, int width, int height,int arcWidth,int arcHeight)

  is used to fill rounded rectangle with the default color and specified width and height.

- public void setColor(Color c)

  is used to set the graphics current color to the specified color.

- public void setFont(Font font)

  is used to set the graphics current font to the specified font.

Displaying Graphics in Applet

- public void drawLine(int x1, int y1, int x2, int y2)

  is used to draw line between the points(x1, y1) and (x2, y2).

- public void drawArc(int x, int y, int width, int height, int startAngle, int arcAngle)

  is used draw a circular or elliptical arc.

- public void fillArc(int x, int y, int width, int height, int startAngle, int arcAngle)

  is used to fill a circular or elliptical arc.

- public void showstatus(“string to display”)

  is used to display status

setBackground and setForeground

- To set the color of the background of an applet window, setBackground () method is used.
- The general form of the setBackground () method is

  void setBackground(mycolor)

- Similarly, to set the foreground color to a specific color, that is, the color of text, setForeground () method is used.
- The general form of the setForeground () method is

  void setForeground(mycolor)

- mycolor is one of the color constants or the new color created by the user
- The list of color constants is given below:

  • Color.red • Color.orange • Color.gray • Color.darkGray • Color.lightGray • Color.cyan •Color.pink • Color.white • Color.blue • Color.green • Color.black • Color.yellow

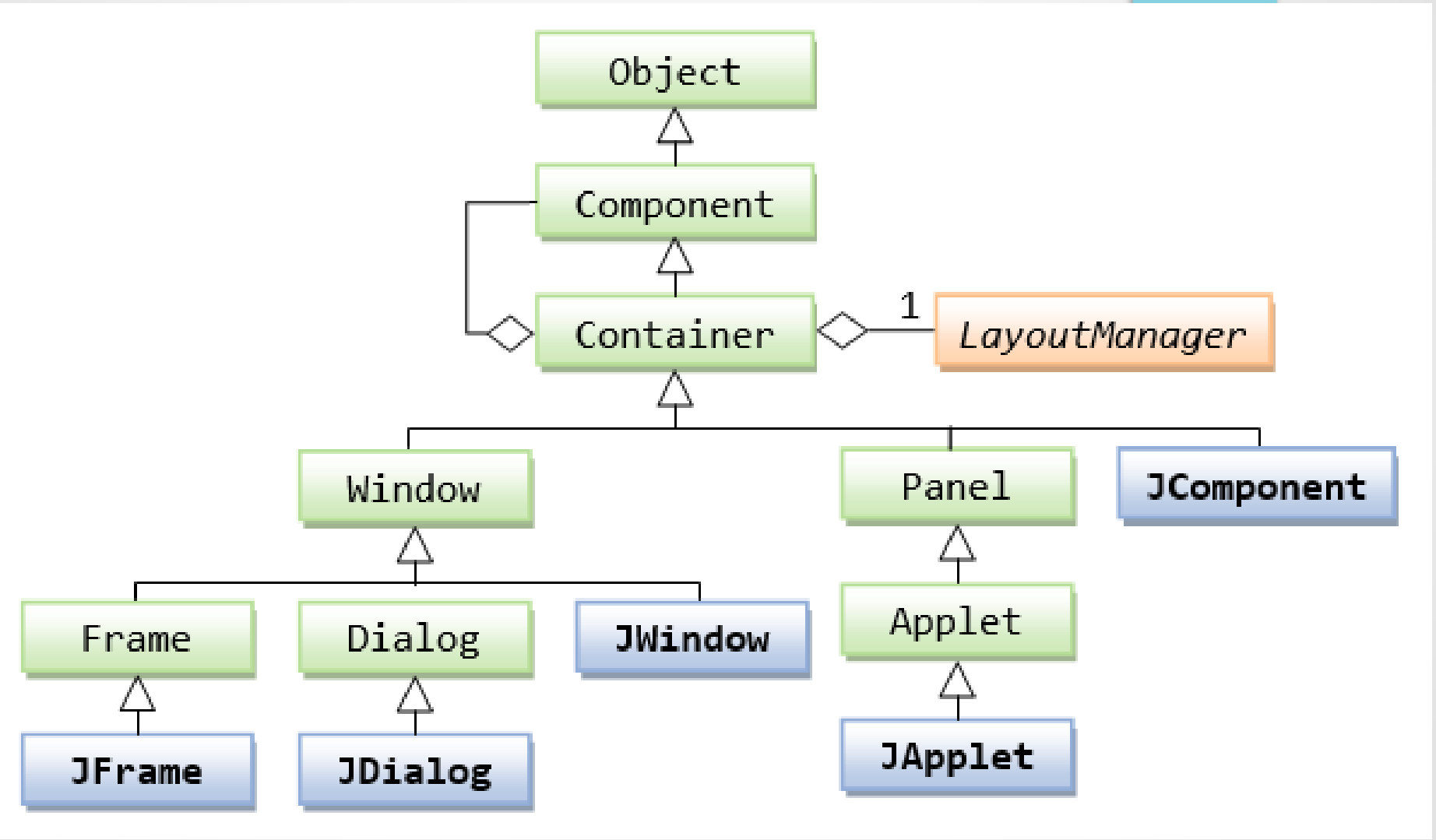
Difference between Applet and JApplet

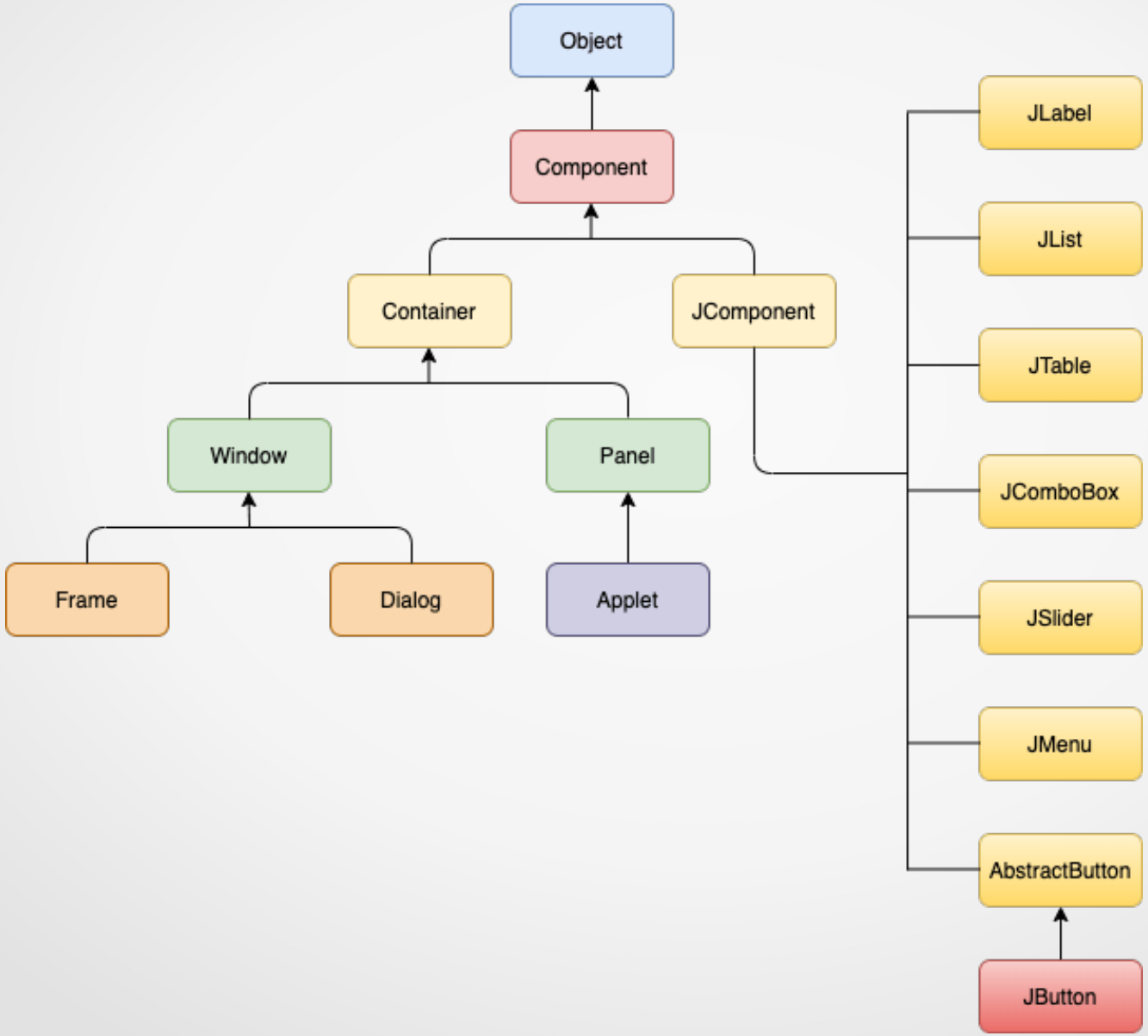
- To create an applet, the user can import either the Applet class (to write an AWT applet) or the JApplet class (to write a Swing applet)
- The Applet class is in java.awt package
- The Japplet class is in javax.swing package
- Both Applet and JApplet contain code that works with a browser to create a display window

JApplet

- The javax.swing.JApplet class defines the core functionality of an applet. (java.awt.Applet is the older, AWT-based form.)
- Initially, Java introduced a package called **Abstract Window Took Kit (AWT).**
- This package contained a large number of classes and interfaces that supported the creation of GUI.
- Newer version of this package in Java2 is called **Swing.**
- The Swing classes are a part of the **Java Foundation Classes (JFC).**
- The Swing classes are contained in a Java extension package called **javax.**

Swing


Object
Component
Container
1
LayoutManager
Window
Panel
JComponent
Frame
Dialog
JWindow
Applet
JFrame
JDialog
JApplet


Object
Component
Container
JComponent
Window
Panel
Frame
Dialog
Applet
JLabel
JList
JTable
JComboBox
JSlider
JMenu
AbstractButton
JButton

JFrame

- A Jframe window is a standard style window.
- **JFrame is a subclass of Frame class.**
- When **JFrame** is created, its size is (0,0) and is **invisible**
- A **JFrame** Methods:
  - *String getTitle()*
  - *void setTitle(String title)*
  - *void setVisible(boolean b)*
  - *void setSize(int width, int hight)*

JPanel

- The JPanel is a simplest container class.
- It provides space in which an application can attach any other component. It inherits the JComponents class.
- It doesn't have title bar.
- The main task of JPanel is to organize components.
- Methods:
  - add(Component c)

JLabel

- Jlabel is a built in Java Swing class that holds text you can display within an applet.
- Jlabel class is concrete subclass of **Jcomponent.**
- Methods:
  - String getText()
  - void setText(String str)

JButton

- The JButton is a conrete subclass of abstract Button which is a sub class of Jcomponent.
- The JButton class is used to mouse press and mouse release events can be precessed separately.
- Methods:
  - String getText()
  - void setText(String str)

JTextField

- Simple type texts are dealt by JtextField, JpasswordField and JtextArea classes
- JtextField can display one line of editable text of one font and color at a time.
- The object of a JTextField class is a text component that allows the editing of a single line text.

JTextArea

- The object of a JTextArea class is a multi line region that displays text.
- It allows the editing of multiple line text.
- It inherits JTextComponent class

JCheckBox

- The JCheckBox class is used to create a checkbox. It is used to turn an option on (true) or off (false).
- Clicking on a CheckBox changes its state from "on" to "off" or from "off" to "on ".
- It inherits JToggleButton class.

JComboBox

- The object of Choice class is used to show popup menu of choices. Choice selected by user is shown on the top of a menu.
- It inherits JComponent class.
- Methods:
  - void addItem(Object anObject)
  - void removeItem(Object anObject)

JRadioButton

- The JRadioButton class is used to create a radio button.
- It is used to choose one option from multiple options.
- It is widely used in exam systems or quiz.
- Methods:
  - void setText(String s)
  - String getText()