Object Oriented Programming in Java

Er.Sital Prasad Mandal

BCA- 3th sem
Mechi Multiple Campus
Bhadrapur, Jhapa, Nepal
(Email : info.sitalmandal@gmail.com)
https://ctaljava.blogspot.com/

Text Book

1. Deitel & Dietel. -Java: How to-program-. 9th Edition. TearsorrEducation. 2011, ISBN: 9780273759168

2. Herbert Schildt. "Java: The CoriviaeReferi4.ic e 61 Seventh Edition. McGraw -Hill 2006, 1SBN; 0072263857

Introduction to Java Applets

1. Definition
2. Applet lifecycle methods
3. Build a simple applet
4. Using Applet Viewer
5. Adding Controls: Animation Concepts.

Introduction to Java Applets

Definition:

- A Java program that runs in a Web browser is called **APPLET.**
- Applet is a special type of program that is embedded in the webpage to generate the dynamic content.

Introduction to Java Applets

Applet	Application
1. Lines of code is very less - Small Program	1. Lines of coding are usually large - Large Program
2. Act as a client program :- used to run a program on client Browser	2. Executed on stand alone computer system
3. Applet is portable and can be executed by any JAVA supported browser such as Mozilla, chrome, IE	3. Need JDK, JRE, JVM installed on client machine.
4. Applet applications are executed in a Restricted Environment	4.Application can access all the resources of the computer

Introduction to Java Applets

5. Applets are created by extending the java.applet.Applet	5. Applications are created by writing public static void main(String[] s) method.
6. Applet application has 5 methods which will be automatically invoked event handling process.	6. Application has a single start point which is main method

Example:
import java.awt.*;
import java.applet.*;
public class <class-name> extends Applet
{
public void init() { }
public void start() {}
public void stop() {}
public void destroy() {}
public void paint(Graphics g) {}
}
	Example :
	class <Class-name>
	{
	public static void main(String args[])
	{
	System.out.println(“Welcome”);
	}
	}

Applet Lifecycle Methods

- Applet has 4 life cycle methods such as

1. init()
2. Start()
3. Stop()
4. Destroy()

Applet Lifecycle Methods

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.

Applet Lifecycle Methods

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.

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

Applet Lifecycle Methods

Who is responsible to manage the life cycle of an applet?

Java Plug-in software.

How to run an Applet?

There are two ways to run an applet

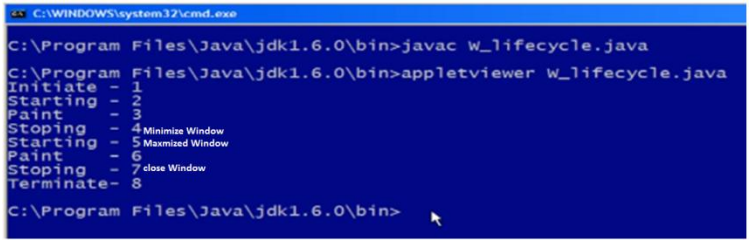
- By html file.
- By appletViewer tool (for testing purpose).

Applet Lifecycle Methods

Paint()

- Public void paint(Graphics g) – used to paint the applet.
- Java plugin software – manage the applet

Applet Lifecycle Methods




```
import java.applet.Applet;
import java.awt.*;
public class AppletLifeCycle extends Applet {
    int count =1;
    public void init () {
        System.out.println("initiate" + count ++);
    }
    public void start () {
        System.out.println("start" + count ++);
    }
    public void stop () {
        System.out.println("stop" + count ++);
    }
    public void destroy () {
        System.out.println("destroy" + count ++);
    }
    public void paint ( Graphics g ) {
        System.out.println("Paint - " + count ++);
        g.drawString("Welcome to Java Applet !", 40 , 60);
    }
}
/*<applet code="AppletLifeCycle.class" width="400", height="250">
</applet>*/
```

CMD

/>Appletviewer Applet.html

Simple Applet program

```
import java.applet.*;
import java.awt.Graphics;
 public class firstapplet extends Applet
{
public void paint(Graphics g)
{
g.drawOval(100,200,300,100);
}
}
/*  <applet code="firstapplet.class" width="300"
height="300">
    </applet>  */
```

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 abstract 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 abstract void drawOval(int x, int y, int width, int height):** is used to draw oval with the specified width and height.
- **public abstract void fillOval(int x, int y, int width, int height):** is used to fill oval with the default color and specified width and height.
- **public abstract void drawLine(int x1, int y1, int x2, int y2):** is used to draw line between the points(x1, y1) and (x2, y2).

Commonly used methods of Graphics class:

- **public abstract boolean drawImage(Image img, int x, int y, ImageObserver observer):** is used draw the specified image.
- **public abstract void drawArc(int x, int y, int width, int height, int startAngle, int arcAngle):** is used draw a circular or elliptical arc.
- **public abstract void fillArc(int x, int y, int width, int height, int startAngle, int arcAngle):** is used to fill a circular or elliptical arc.
- **public abstract void setColor(Color c):** is used to set the graphics current color to the specified color.
- **public abstract void setFont(Font font):** is used to set the graphics current font to the specified font.

Applet Example

```
import java.applet.Applet;
import java.awt.Color;
import java.awt.Graphics;
public class GraphicDemo extends Applet {
    public void paint(Graphics g) {
        g.drawString("Welcome", 50, 50);
        g.drawLine(20, 30, 20, 300);
        g.drawRect(70, 100, 30, 30);
        g.fillRect(170, 100, 30, 30);
        g.drawOval(70, 200, 30, 30);
        g.setColor(Color.red);
        g.fillOval(170, 200, 30, 30);
        g.drawArc(90, 150, 30, 30, 30, 270);
        g.fillArc(270, 150, 30, 30, 0, 180);
    }
}
```

Display the image in Applet

- Applet is mostly used in games and animation. For this purpose image is required to be displayed. The java.awt.Graphics class provide a method drawImage() to display the image.

Syntax of drawImage() method:

public abstract boolean drawImage(Image img, int x, int y, ImageObserver observer):

is used draw the specified image.

How to get the object of Image:

- The java.applet.Applet class provides getImage() method that returns the object of Image.

Syntax:

public Image getImage(URL u, String image){}

How to get the object of Image:

Other required methods of Applet class to display image:

- public URL getDocumentBase(): is used to return the URL of the document in which applet is embedded.

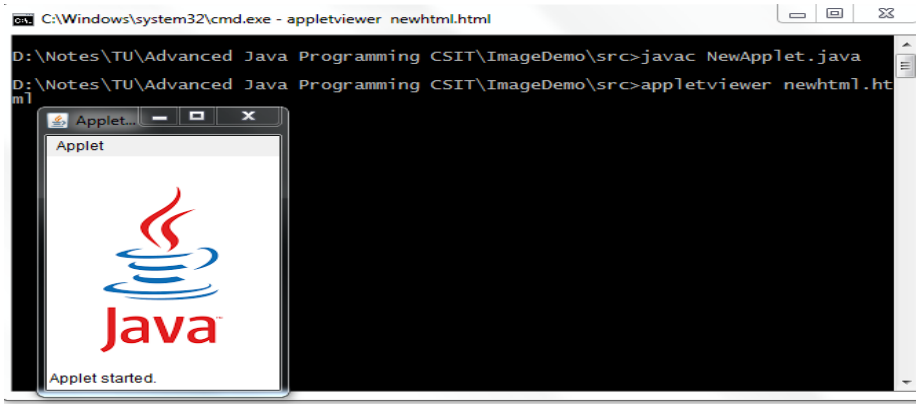
- public URL getCodeBase(): is used to return the base URL.

How to get the object of Image:

```
import java.awt.*;
import java.applet.*;

public class NewApplet extends Applet {
    Image picture;
    public void init() {
        picture = getImage(getDocumentBase(), "javalogo.jpg");
    }

    public void paint(Graphics g) {
        g.drawString("Image", 50, 50);
        g.drawImage(picture, 30, 30, this);
    }
}
/*
 <applet code="image.class" width="600" height="600">
 </applet>
 */
```




```
D:\ImageDemo\src>javac NewApplet.java
D:\ImageDemo\src>appletviewer newhtml.html
```

Animation in Applet

```
2  import java.awt.*;
3  import java.applet.*;
4  public class animation extends Applet {
5      Image picture;
6      public void init() {
7          picture = getImage(getDocumentBase(), "javalogo.jpg");
8      }
9      public void paint(Graphics g) {
10         for (int i = 0; i < 100; i++) {
11             g.drawImage(picture, i, 30, this);
12
13             try {
14                 Thread.sleep(100);
15             } catch (Exception e) {
16             }
17         }
18     }
19 }
20
21 /*
22 <applet code="animation.class" width="600" height="600">
23 </applet>
24 */
25
```

Animation in Applet

```
1   <!DOCTYPE html>
2   <html>
3       <head>
4           <title>Image</title>
5           <meta charset="UTF-8">
6           <meta name="viewport" content="width=device-width, initial-scale=1.0">
7       </head>
8       <body>
9          <applet code="animation.class" width="600" height="600">
10  </applet>
11      </body>
12  </html>
```

Motivate


We fall. We fail. We break.
But then, we rise. We heal.
We overcome.
@successpictures