SNS COLLEGE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTION)

COIMBATORE – 35

DEPARTMENT OF COMPUTER SIENCE AND ENGINEERING

UNIT V

unit - 5

Designing Java Applets

② major things   AWT → Applet, window Abstract
* Applets ⎫   Toolkid.
* Swing ⎭

window application creation, it has a
separate packages for each.

Applet:
      window based java program used
in internet application. Run this using
applet → viewer / web browser
    (offline)   (online)

There are 2 types of applets:-
      Locale applet → within in a single system
      Remote applet   * No internet required.
            ↓   * Hardware will do the
                          * Operation / Application

* More than one
system connection
* Network should be required.

There must be 2 class in the applet program.
      ⇒ Applet class (Main class) import. Applet.* ;   lang.
      ⇒ Graphic class → lang. awt . * ;

Applet lyf cycle → while implementing the applet
            init () → Variable initialization. (1st step)
  a 1   start () → Call the run method → Run ().
  b 2   stop () → End of execution
  c 3
  d 4   destroy () → Delete the memory occupied
destroy()   during run time.
            paint () → Graphic class must pass.

Structure

```
import java.awt.*;
import java.applet*;
public class class name extends Applet
{
    public void init()
    {
        Initialize variable;
    }
    (override) option is also there
    public void paint (Graphics g)
    {
        g. drawstring (arg),
    }
}
```

Compile process:

① Set path

② compile → javac filename java.

③ Run → appletviewer filnam.java

            ⌐→ ▭

To generat applt tag.

< Applet code = "name class". width = 100 , hight : 100 >

</ Applet >

Swing:
* Portable
* Fixed (56 components)
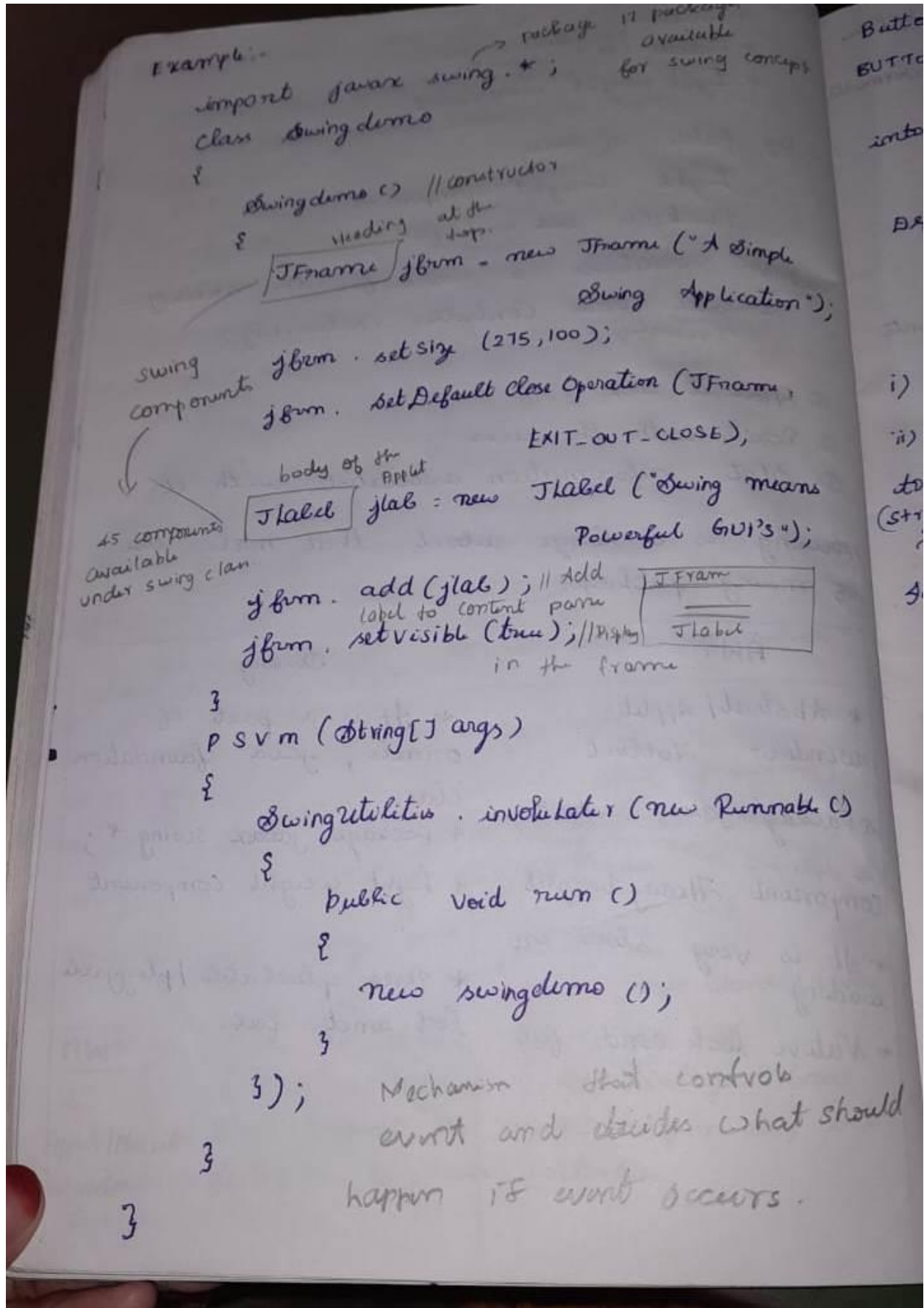* Light weight component

2 key features of swing:-
Light weight component
Pluckable look and feel.

→ MVC connections are used for the visibility.
↳ Model View Controller (Networking)
(Visualize)

① How it looks
② Reacts with the user
③ State information associated with it

→ Swing is a large subset that makes use of many packages.

AWT	Swing (extended version of java)
* Abstract / Applet window Toolkit	* It is a part of oracle, java foundation class.
* Package: java. awt. *; (os dependent) → uses original component Version	* package: javax. swing. *;
component: Heavy weight	* Light weight component.
* It is very slow in loading	* uses pluckable / pluggable look and feel.
* Native look and feel.	

Example:-

→ package if package available
import javax swing .*; for swing concept

class Swingdemo

{

Swingdemo () //constructor

{ Heading at the top.

[JFrame] jfrm = new JFrame ("A Simple Swing Application");

swing components jfrm . set size (275,100);

jfrm . Set Default Close Operation (JFrame, EXIT_OUT_CLOSE),

body of the Applet

[JLabel] jlab = new JLabel ("Swing means Powerful GUI's");

45 components available under swing clan

jfrm . add (jlab); // Add label to content pane

jfrm . setvisible (true);//Display in the frame

[JFram ──── JLabel]

}

p s v m (String[] args)

{

Swingutilities . invokelater (new Runnable ()

{

public void run ()

{

new swingdemo ();

}

}); Mechanism that controls event and decides what should happen if event occurs.

}

}

Example:-

```
import java.awt.*;
import java.applet.*;
Public class basic extends Applet
{
    Public void paint (Graphics g)
    {
        g. drawstring ("Hello Applet",200,200);
    3
    // < applet code = "Basic.class" width = 200, high = 200>
    </applet>
3
```

Compile : javac basic.java.

Run : appletviewer basic.java.

Swing :-

* It is a window based application which is same as awt
* Used Create a Graphical user interface
* Swing is a set of classes that provides more powerful and flexiable GUI components.
* It is a subset of AWT

Only in Java swing was introduced
↳ Java Standard Edition

| AWT | → used to defines basic set of controls, windows and dialogue boxes that support a usable components, but limited to user interface.

↓

Applet / Abstrat window Toolkit.

* Implemented action from actionlistener interface

Listener,
                                      ↓
                       define action performed
                          (report)

Example:-

Java program to print yes, No, decided:-

```java
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
Public class ButtonDemo extends Applet implements
                                        ActionListener
{
    String msg = " ";
    Button yes, no, maybe;
    public void init ()
    {
        yes = new Button ("yes");
        no = new Button ("no");
        maybe = new Button ("undecided");
        add (yes);
        add (no);
        add (maybe);
        yes. addActionListener (this);   // what current object is going on
        no. addActionListener (this);
        maybe. addActionListener (this);
    }
    Public void actionperformed (ActionEvent ae)
    {
        String str = ae. getActionCommand ();
        if (str. equals ("Yes"))
```

Button and mouse Event:

BUTTON EVENT:-

Press button in keyword will converted into object.

Component that contains the label.

Define with 2 constructors for button:-

i) Button () throws Headless Exaption

ii) Button (String str) throws HeadlessException.

i). It is used to create empty button.

ii) Create button but it contain string label to create object.

(string str):
To retrive label by calling get label ()

In program,

Action

Mouse

• Void setlabel (String str)   Focus

• String getlabel ()   Key.

It should handled with some functions

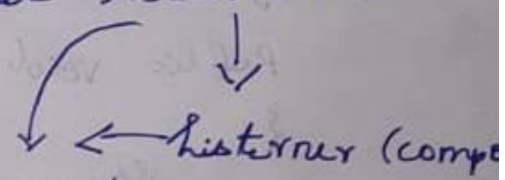HANDLING BUTTONS:-

Specify what action is pressed / released

Action will be taken place while / in process

① Button pressed - Generate Action Event

← Listener (comp

command

(notification) will be recieved

by action event.

EX: downloads from
net.

```
        {
          msg = "you pressed yes.";
        }
        else if (str. equals ("No"))
        {
          msg = "you pressed No.";
        }
        else
        {
          msg = "you pressed undecided.";
        }
        repaint();
      }
                                    add on
      Public void paint (Graphics g)
      {
          g. drawString (msg, 6, 100);
      }
    }
```

MOUSE Event :-

                                    Source (generated event)
MouseEvent (Component Src, int type, action (press
                                                   delet
        long when , int modifers,   modif
              time   visiblity   creat

        int x , int y, coordinate

        int clicks , boolean triggerspopup)
                                    notifications
        Button   (Enable)
        pressed.