Page No :
Date :

Applet

Functions

init()
start()
paint()
stop()
destroy()

Applet is a window

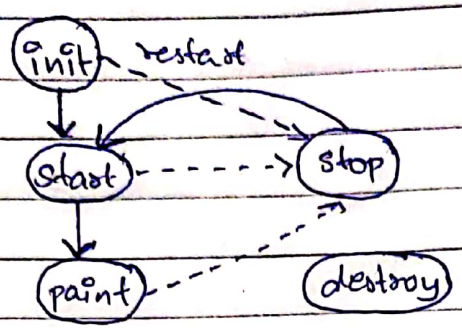
```
import java.applet.*;
import java.awt.*;
class Test extends Applet {
    Public void paint(Graphics g) {
        g.drawString("Hello World", 40,50);
    }
}
/*
<html>
<applet code = "Test.class" width=400 height=500>
</applet>
</html>
*/
```

coordinates

Scanned by CamScanner

Page No :
Date :

Life cycle of an applet.




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

/*
<html>
<applet code="Test.class" width=400 height=500>
</applet>
</html>
*/
public class Test extends Applet {
        private String msg = " ";
        public void init() {
                msg = msg + "Inside init()";
        }
        public void start() {
                msg = msg + "--> Inside Start()";
        }
        public void paint (Graphics g){
                msg = msg + "--> Inside paint()";
                g.drawString (msg, 50, 100);
        }
```

Scanned by CamScanner

```
g.drawRect(100,120,80,80)

public void stop(){
        msg = msg + " --> Inside stop()";
}
public void destroy(){
        msg = msg + " --> Inside destroy()";
}
}
```

Output

Applet.
Inside init() --> Inside start --> inside point()
Applet started.

Q] Write an applet program to draw a rectangle and fill the rectangle with green colour.

```
import java.applet.*;
import java.awt.*;
/*
<html>
<applet code="Test.class" width=400 height=500>
</applet>
</html>
*/
public class Test extends Applet{
        public void paint(Graphics g){
                g.drawRect(100,120,150,70);
                g.setColor(color.green);
                g.fillRect(100,120,150,70);
```

Scanned by CamScanner

Page No :
Date :

Q] Write a java program to add 2 numbers and display the sum. Assume the numbers are sent to Applet using parameters.

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

/*
<html>
<applet code="Test.class" width=400 height=500>
<param name="num1" value=15/>
<param name="num2" value=25/>
</applet>
</html>
*/
public class Test extends Applet {
    public void paint(Graphics g){
        int a = Integer.parseInt(getParameter("num1"));
        int b = Integer.parseInt(getParameter("num2"));
        int sum = a+b;
        g.drawString("Sum="+sum,100,200);
    }
}
```

Scanned by CamScanner

8] write a java program to find factorial of a number using applet. Assume the no. is passed as an argument.

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

/*
<html>
<applet code="Test.class" width=400 height=500>
<param name="num" value=5 />
</applet>
</html>
*/
public class Test extends Applet {
    public int fact(int n) {
        if (n==0)
            return 1;
        return n*fact(n-1);
    }
    public void paint(Graphics g){
        int n = Integer.parseInt(getParameter("num"));
        g.drawString("Factorial = "+fact(n),100,200);
    }
}
```

Scanned by CamScanner

Date :

Q] write an applet program to perform Arithmetic operation

```
import java.applet.* ;
import java.awt.* ;
/*
<html>
<applet code = " Test.class" width = 400 height = 500 >
<param name = " num1 " value = 5 />
<param name = " num2" value = 10 />
</applet >
</html>
*/
public class Test extends Applet {
    public void paint (Graphics g) {
        int a = Integer.parseInt(getParameter("num1"));
        int b = Integer.parseInt(getParameter("num2"));
        int sum = a+b;
        int diff = a-b;
        int mul = a*b;
        int div = a/b;
        int mod = a%b;
        g.drawString ("sum = "+ sum , 100, 200);
        g.drawString ("diff = " + diff , 100, 230);
        g.drawString ("mul = "+ mul , 100, 260);
        g.drawString ("div = "+ div , 100, 290);
        g.drawString ("mod = "+ mod , 100, 320);
    }
}
```

Scanned by CamScanner

Page No :
Date :

Q] Write a Java prgm to add two numbers by taking values from user using applet.

```
import java.applet.*;
import java.awt.*;          ← (textfield, label, button)
import java.awt.event.*;    ← (actionListener)

public class Test extends Applet implements ActionListener{

    private Label l1;
    private Label l2;
    private Label l3;

    private TextField t1;
    private TextField t2;
    private TextField t3;

    private Button b1;

    public void start() {
        l1 = new Label("Enter Number 1");
        l2 = new Label("Enter Number 2");
        l3 = new Label("Result");

        t1 = new TextField(10);
        t2 = new TextField(10);
        t3 = new TextField(20);

        b1 = new Button("Submit");
```

Scanned by CamScanner

```
        add (l1); add (t1);
        add (l2); add (t2);
        add (b1);
        add (l3); add (t3);

        b1.addActionListener (this);
    }
    public void actionPerformed (ActionEvent e) {
        Integer result = Integer.parseInt(t1.getText())
                + Integer.parseInt(t2.getText());
        t3.setText( result.toString());
    }
}
```

Q] WAJ to convert velocity in km/h to m/s

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

public class Test extends Applet implements ActionListener {

    private Label l1;
    private Label l2;

    private TextField T1;
    private TextField T2;

    private Button b1;
```

Scanned by CamScanner

Page No :
Date :

```
public void start () {
    l1 = new Label ("Enter velocity in km/h");
    l2 = new Label ("velocity in m/s");

    t1 = new TextField (10);
    t2 = new TextField (10);

    b1 = new button ("convert");

    add (l1); add (t1);
    add (b1);
    add (l2); add (t2);

    b1 . addActionListener (this);
}
public void actionPerformed (ActionEvent e) {
    Double result = Double . parseDouble (t1 . getText()
                    * 5.0/18
    t2 . setText (result . toString ());
}
}
```

Scanned by CamScanner

Q] WAS to perform arithmetic operations.

```
import java.applet.*
import java.awt.*
import java.awt.event.*

public class Test extends Applet implements ActionListener {

        private Label l1;
        private Label l2;
        private Label l3;

        private TextField t1;
        private TextField t2;
        private TextField t3;

        private Button b1;
        private Button b2;
        private Button b3;
        private Button b4;

        public void start () {
                l1 = new label ("Enter no. 1");
                l2 = new label ("Enter no. 2");
                l3 = new Label ("Result");

                t1 = new TextField (10);
                t2 = new TextField (10);
                t3 = new TextField (20);
```

Scanned by CamScanner

e.getSource()

Page No:
Date:

```
b1 = new Button("Add");
b2 = new Button("subtract");
b3 = new Button("Multiplication");
b4 = new Button("Division");

add(l1); add(t1);
add(l2); add(t2);
add(b1); add(b2); add(b3); add(b4);
add(l3); add(t3);

b1.addActionListener(this);
b2.addActionListener(this);
b3.addActionListener(this);
b4.addActionListener(this);
}
public void actionPerformed(ActionEvent e){
    Integer n1 = Integer.parseInt(t1.getText());
    Integer n2 = Integer.parseInt(t2.getText());

    if (e.getSource() == b1){
        Integer result = n1 + n2;
        t3.setText(result.toString());
    }
    else if (e.getSource() == b2){
        Integer result = n1 - n2;
        t3.setText(result.toString());
    }
    else if (e.getSource() == b3){
        Integer result = n1 * n2;
        t3.setText(result.toString());
    }
```

Scanned by CamScanner

```
        else {
            Integer result = n1 / n2;
            t3.setText(result.toString());
        }
    }
}
```

Q] WAJP to reverse a string using applet.

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

/*
<html>
<applet code="Test.class" width=600 height=500>
</applet>
</html>
*/

public class Test extends Applet implements ActionListener{

    private Label l1;
    private Label l2;

    private TextField t1;
    private TextField t2;

    private Button b1;
```

Scanned by CamScanner

CPP	Java
s[2]	s.charAt(2)
strlen()	s.length()
strcmp()	s.equals()

Page No :
Date :

charAt()
length()
equals()
compare()
toString()

```
public void start (){
    l1 = new Label (" Enter string");
    l2 = new Label (" Reversed string");

    t1 = new TextField (10);
    t2 = new TextField (10);

    b1 = new Button ("Reverse");

    add (l1); add (t1);
    add (b1);
    add (l2); add (t2);

    b1.addActionListener (this);
}
public void ActionPerformed (ActionEvent e){

    String str = t1.getText();
    String rev = " ";
    for (int i = str.length() - 1 ; i >= 0 ; i--)
        rev = rev + str.charAt(i);
    t2.setText (rev);
}
}

/* code for palindrome.
    if (rev.equals (str)){
        t2.setText (" Palindrome");
    }
    else
        t2.setText ("not palindrome");

*/
```

Scanned by CamScanner

Q] WAJP to find no. of vowels, consonents, digits in a given string using applet.

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

public class Test extends Applet implements ActionListener {

    private Label l1;
    private Label l2;
    private Label l3;
    private Label l4;

    private TextField t1;
    private TextField t2;
    private TextField t3;
    private TextField t4;

    private Button b1;

    public void start() {
        l1 = new Label ("Enter String:");
        l2 = new Label ("no. of vowels");
        l3 = new Label ("no. of consonents");
        l4 = new Label ("no. of digits");

        t1 = new TextField (10);
        t2 = new TextField (10);
        t3 = new TextField (10);
        t4 = new TextField (10);
```

Scanned by CamScanner

ASCII		
	65 → A	97 → a
	90 → Z	121 → z
	48 → 0	57 → 9

Page No :
Date :

```
        b1 = new Button ("Submit");

        add (l1); add (t1);
        add (b1);
        add (l2); add (t2);
        add (l3); add (t3);
        add (l4); add (t4);

        b1.addActionListener (this);
    }

    public void actionPerformed (ActionEvent e) {
        String str = t1.getText();
        Integer vow = 0;
        Integer con = 0;
        Integer dig = 0;

        for (int i=0; i<str.length(); i++) {
            char c = str.charAt(i);
            if (c = 'A' || c = 'E' || c = 'I' ||
                ------ c = 'u') {
                vow++;
            }
            else if (c >=65 && c <=90 || c>=97
                                && c <=121) {
                con++;
            else if (c>=48 && c <=57)
                dig++;
        }
        t2.setText (vow.toString());
        t3.setText (con.toString());
        t4.setText (dig.toString());
    }
}
```

Scanned by CamScanner

Page No :
Date :

SWING

Q] Concate two strings.

```
import javax.swing.*;
import java.awt.event.*;
class Concate extends JFrame implements ActionListener{
        private JLabel l1;
        private JLabel l2;
        private JLabel l3;

        private JTextField t1;
        private JTextField t2;
        private JTextField t3;

        private JButton b1;

        public concate (){
                setSize (400,500);
                setLayout (null);
                l1 = new JLabel ("string 1");
                l2 = new JLabel ("String 2");
                l3 = new JLabel ("Result");

                t1 = new JTextField (10);
                t2 = new JTextfield (10);
                t3 = new JTextField (10);

                b1 = new JButton ("Submit");
```

Scanned by CamScanner

```
        l1.setBounds(100, 100, 50, 30);
        l2.setBounds(100, 140, 50, 30);
        l3.setBounds(100, 220, 50, 30);

        t1.setBounds(180, 100, 150, 30);
        t2.setBounds(180, 140, 150, 30);
        t3.setBounds(180, 220, 150, 30);

        b1.setBounds(140, 180, 100, 30);

        add(l1); add(t1); add(l2); add(t2);
        add(b1);
        add(l3); add(t3);

        b1.addActionListener(this);
    }

    public void actionPerformed(ActionEvent e){
        t3.setText(t1.getText() + t2.getText());
    }

    public static void main(String args[]){
        concate c = new concate();
        c.setVisible(true);
    }
}
```

Scanned by CamScanner

$F = C * 9/5 + 32$ $\quad (32 \times F - 32) \times \frac{5}{9} =$

Page No :
Date :

Q] convert temp in °C to °F using swing.

```
import javax.swing.*;
import java.awt.event.*;

public class Test extends JFrame implements ActionListener {
        private JLabel l1;
        private JLabel l2;

        private JTextField t1;
        private JTextField t2;

        private JButton b1;

        public Test() {
            setSize(400, 500);
            setLayout(null);
            l1 = new JLabel("temp in celsius");
            l2 = new JLabel("temp in Fahrenheit");

            t1 = new JTextField(10);
            t2 = new JTextField(10);

            b1 = new JButton("convert");

            l1.setBounds(100, 200, 50, 50);
            l2.setBounds(100, 250, 50, 50);
            t1.setBounds(200, 200, 50, 50);
            t2.setBounds(200, 250, 50, 50);
            b1.setBounds(150, 300, 50, 50);

            add(l1); add(l2); add(t1); add(t2); add(
```

Scanned by CamScanner

```
        b1.addActionListener(this);
    }
    public void actionPerformed(ActionEvent e){
        Double C = Double.parseDouble(t1.getText());
        Double F = (C * 9.0 / 5.0) + 32
        t2.setText(F.toString());
    }
    public static void main(String args[]){
        Test t = new Test();
        t.setVisible(true);
    }
}
```

Q] Arithmetic operations using swing.

```
import javax.swing.*;
import java.awt.event.*;

public class Test extends JFrame implements ActionListener{
    private JLabel l1;
    private JLabel l2;
    private JLabel l3;

    private JTextField t1;
    private JTextField t2;
    private JTextField t3;

    private JButton b1;
    private JButton b2;
    private JButton b3;
    private JButton b4;
```

Scanned by CamScanner

```
public Test () {
    setSize (400, 500);
    setLayout (null);

    l1 = new JLabel (" Num 1 ");
    l2 = new JLabel (" Num 2 ");
    l3 = new JLabel (" Result ");
    t1 = new JTextField (10);
    t2 = new JTextField (10);
    t3 = new JTextField (20);
    b1 = new JButton (" Add ");
    b2 = new JButton (" Sub ");
    b3 = new JButton (" Mul ");
    b4 = new JButton (" Div ");

    l1.setBounds (100, 100, 50, 50);
    l2.setBounds (100, 150, 50, 50);
    l3.setBounds (100, 200, 50, 50);
    t1.setBounds (150, 100, 50, 50);
    t2.setBounds (150, 150, 50, 50);
    t3.setBounds (150, 200, 50, 50);
    b1.setBounds (200, 100, 50, 50);
    b2.setBounds (220, 120, 50, 50);
    b3.setBounds (240, 140, 50, 50);
    b4.setBounds (260, 160, 50, 50);

    add (l1); add (l2); add (l3);
    add (t1); add (t2); add (t3);
    add (b1); add (b2); add (b3); add (b4
```

Scanned by CamScanner

Page No :
Date :

```
        b1.addActionListener (this);
        b2.addActionListener (this);
        b3.addActionListener (this);
        b4.addActionListener (this);
    }
    public void actionPerformed (ActionEvent e){
        Double n1 = Double.parseDouble (t1.getText());
        Double n2 = Double.parseDouble (t2.getText());

        if (e.getSource == b1){
            Double result = n1 + n2;
            t3.setText(result.toString());
        }
        else if (e.getSource == b2){
            Double result = n1 - n2;
            t3.setText(result.toString());
        }
        else if (e.getSource == b3){
            Double result = n1 * n2;
            t3.setText(result.toString());
        }
        else if (e.getSource == b4){
            Double result = n1/n2;
            t3.setText(result.toString());
        }
    }
    public static void main (String args[]){
        Test t = new Test();
        t.setVisible (true);
    }
```

Scanned by CamScanner

mouseEntered.
mouse Released.
mouse Pressed.
mouseClicked

Page No :
Date :

Q] WAJP that will explain the working of mouse event handling.

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

public class MouseTest extends Applet implements MouseListener {

    public void start() {
        addMouseListener(this);
    }
    public void mouseExited(MouseEvent e) {
        setBackground(Color.red);
    }
    public void mouseEntered(MouseEvent e) {
        setBackground(Color.green);
    }
    public void mouseReleased(MouseEvent e) {
        setBackground(Color.blue);
    }
    public void mousePressed(MouseEvent e) {
        setBackground(Color.yellow);
    }
    public void mouseClicked(MouseEvent e) {
        setBackground(Color.cyan);
    }
}
```

Scanned by CamScanner

mouse Moved
mouse Dragged

Page No :
Date :

8] WAJP to display the coordinates of a mouse motion.

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

public class MouseTest extends Applet implements
                                MouseMotionListener {
        private int x;
        private int y;

        public void start(){
            addMouseMotionListener(this);
        }
        public void mouseMoved(MouseEvent e){
            x=e.getX();
            y=e.getY();
            repaint();
        }
        public void mouseDragged(MouseEvent e){
            x=e.getX();
            y=e.getY();
            repaint();
        }
        public void paint(Graphics g){
            showStatus("("+x+","+y+")");
            g.drawString("("+x+","+y+")",x,y);
        }
}
```

to show in status bar → showStatus("("+x+","+y+")");

to show with mouse → g.drawString("("+x+","+y+")",x,y);

Scanned by CamScanner

url = " jdbc : oracle : thin : @ localhost : 1521 : xe " ;

IP address = @localhost. 127.0.0.0

port No. : 1521
schema = xe
Username = system
any password.

Driver → thin.

```
import java.sql.*;
class Test{
    PSVM (String args[]){
        try {
            Class.forName("oracle.jdbc.driver.OracleDriver");
```

checking class exists.

Package name. Class Name.

```
            String url = " jdbc : oracle : thin : @localhost : 1521 : xe " ;
            String name = "system";
            String password = " password";
            Connection con = DriverManager.getConnection
                                (url, name, password);

            Statement stmt = con.createStatement();

            String sql = "select * from Person";
            Resultset rs = String executeQuery(sql);
            while (rs.next());{
                SOP(rs.getString(1) + "\t" + rs.getInteger(2)
                    + "\t" + rs.getString(3) + "\t" + rs.getString(4));
            }
            con.close();
    }
```

Scanned by CamScanner

insert
create } executeUpdate()
update

select → executeQuery()

```
catch (SQLException e){
        sop(e.getMessage());
}
catch (ClassNotFoundException e){
        sop(e.getMessage());
}
}
}
}
```

ojdbc 6-~~[illegible]~~jar

1) Creation of statement.
2) Execution of statement.

Library
↳ right click
↳ add Jar
↳ select ojdbc 6 jar

File
project. → name
source package.
rest default package.
new java Class
Class name. DB connect
Package com.s4.

right click → run.

Scanned by CamScanner

Q] Create a Java pgm that will display all the details of book table that contain following attribute.
BookID, Name, AuthorName, NoOfPages
Also that will update the no. of pages of BookID 121 to 3000.

```
import java.sql.*;

public class Test {
        PSVM (String args[]) {
            try {
                Class.forName("oracle.jdbc.driver.OracleDriver");
                String url = "jdbc:oracle:thin:@localhost:1521:xe";
                String name = "System";
                String password = "password";
                Connection con = DriverManager.getConnection(url,
                                                    name, password);

                Statement stmt = con.createStatement();
                String sql = "select * from Book";
                ResultSet rs = stmt.executeQuery(sql);
                while(rs.next()) {
                    System.out.println(rs.getInt(1) +
                        "\t" + rs.getName(2) + "\t" + rs.getString(3)
                        + "\t" + rs.getInt(4);
                }
                String sql = "Update Book set NoOfPages = 3000
                                where BookID = 121";
                ResultSet rs = stmt.executeUpdate(sql);
                con.close();
            }
            catch (SQLException e) {
                    SOP(e.getMessage());
            }
            catch (ClassNotFoundException e) {
                    SOP(e.getMessage());
            }
```

Scanned by CamScanner