**Name: Jasani Nikhil DIV: I**

**Roll No: 118 Subject: Java Practical Assignment (403)**

**Index**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr.**  **No** | **Description** | **Page No.** | **Date Of Submission** | **Sign** |
| 1 | Create STUDENT class having data members roll no and name. Create 5 objects of STUDENT class and take input from the user and print all students’ data in ascending order of name with interval of 1 second |  | 15-11-2022 |  |
| 2 | Create an applet which displays a triangle within a circle where the circumference of the circle touches to the all the vertices of the triangle. Provide different colors to both the objects |  | 16-11-2022 |  |
| 3 | Write a Java Program that accepts string data. Extract either All Vowels or All Non-Vowels from given Data According to Options Selection. Also Provide an Option to Display Output in Uppercase or Lowercase. |  | 22-11-2022 |  |
| 4 | Write an applet that simulates a Rotating wheel. User can start and stop rotation by clicking Start and Stop Button. |  | 30-11-2022 |  |
| 5 | Write a Java Program that Accepts String Data from User and then Provide options for Changing case into Any of the Following. (UPPERCASE, lowercase, Sentence case, tOGGLE cASE). |  | 6-12-2022 |  |
| 6 | Write an applet that simulates a Digital Clock displaying in the form Hour:Min:Sec. Users should be able to change the color of the Fonts of the clock by the selection provided. |  | 14-12-2022 |  |
| 7 | Write a program to draw Smiley face with color using applet |  | 20-12-2022 |  |
| 8 | Write a program that accept Book information like Title, Author, Publication and Price for the N book from the user and display books in descending order with interval of 1 second using thread. |  | 27-12-2022 |  |
| 9 | Writer a program to demonstrate general operations of circular link list using switch case. |  | 3-01-2023 |  |
| 10 | Writer a program to demonstrate general operations of singly link list using switch case. |  | 11-01-2023 |  |
| 11 | Write a program which demonstrate sunset using an Applet. |  | 17-01-2023 |  |
| 12 | Write a program add/sub/mul/div of 2 numbers in Applet using button. |  | 24-01-2023 |  |
| 13 | Accept N number from applet tag and print that many numbers of object using an Applet. |  | 31-01-2023 |  |
| 14 | Write a program which demonstrate functionality of creation and use of package. |  | 08-02-2023 |  |
| 15 | Accept number from applet tag parameter and draw that many numbers of objects. Fill them with different color. |  | 15-02-2023 |  |
|  |  |  |  |  |

**Date:**

**Assign mentor’s Signature:**

**Ms. Rupal Panchal**

**Programmme 1:**

**create STUDENT class having data members no and name. Create 5 objects of STUDENT class take input from the user and print all students' data and in ascending order of name with interval of 1 second.**

import java.util.Scanner;

import java.io.\*;

import java.lang.Thread;

public class stud extends Thread

{

int rno, s;

int n[] = new int[5];

String q[] = new String[5];

String name, p;

public void get()

{

for(int i=0; i<5; i++)

{

System.out.println("Enter Rno: ");

Scanner s = new Scanner(System.in);

n[i] = s.nextInt();

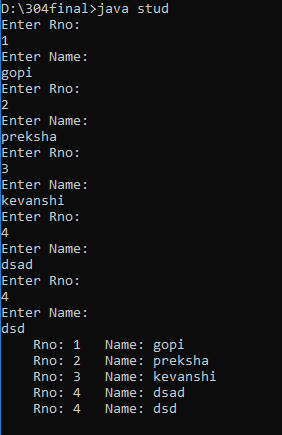
System.out.println("Enter Name: ");

Scanner p = new Scanner(System.in);

q[i] = p.nextLine();

}

}

 public void put()

{

for(int i=0; i<5; i++)

{

Thread t = new Thread(this);

System.out.print(" Rno: n[i]);

System.out.println(" Name:+q[i]);

try{

t.sleep(1000);

}{}

catch(InterruptedExceptione }

}

public static void main(String args[])

{

stud q1 = new stud();

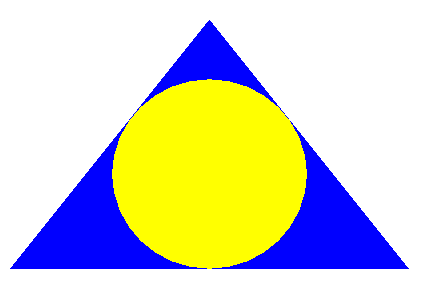
q1.get();

q1.put();

} };

**Programmme 2:**

**Create an applet which displays a Triangle within a circle where the circumference of the circle touches to the all the vertices of the triangle. Provide different colors to both the objects.**



/\* <applet code="que2" height=500 width=700></applet> \*/

import java.awt.\*;

import java.applet.\*;

public class que2 extends Applet

{

public void paint(Graphics g)

{

//Triangle

g.setColor(Color.blue);

int x[] = {350, 150, 550};

int y[] = {100, 350, 350};

int n = 3;

g.fillPolygon(x, y, n);

//oval

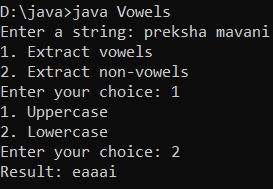
g.setColor(Color.yellow);

g.fillOval(253, 160, 195, 190); } };

**Programme 3:**

**:Write a Java Program that accepts string data.Extract either All Vowels or All Non-Vowels from givenData According to Options Selection. Also Provide anOption to Display Output in Uppercase or Lowercase.**

import java.util.Scanner;

public class Vowels {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a string: ");

String str = sc.nextLine();

System.out.print("1. Extract vowels\n2. Extract non-vowels\nEnter your choice: ");

int choice = sc.nextInt();

sc.nextLine();

System.out.print("1. Uppercase\n2. Lowercase\nEnter your choice: ");

int caseChoice = sc.nextInt();

sc.close();

String vowels = "aeiouAEIOU";

String result = "";

for (int i = 0; i < str.length(); i++) {

char c = str.charAt(i);

if (choice == 1 && vowels.indexOf(c) != -1 || choice == 2 && vowels.indexOf(c) == -1) {

result += c;

}

}

if (caseChoice == 1) {

result = result.toUpperCase();

} else {

result = result.toLowerCase();

}

System.out.println("Result: " + result);

} }

**Programme 4:**

**Write an applet that simulates a Rotating wheel. User can start and stop rotation by clicking Start andStop Button.**

importjava.awt.\*;

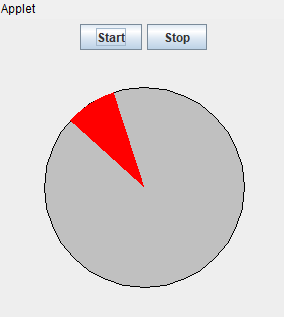
importjava.awt.event.\*;

importjavax.swing.\*;

/\*<applet code="RotatingWheel.class" height=300 width=300></applet>\*/

public class RotatingWheel extends JApplet implements ActionListener {

privateJButtonstartButton;

privateJButtonstopButton;

privateWheelComponent wheel;

private Timer timer;

public void init() {

Container content = getContentPane();

content.setLayout(new BorderLayout());

startButton = new JButton("Start");

stopButton = new JButton("Stop");

startButton.addActionListener(this);

stopButton.addActionListener(this);

JPanel control = new JPanel();

control.add(startButton);

control.add(stopButton);

content.add(control, BorderLayout.NORTH);

wheel = new WheelComponent();

content.add(wheel, BorderLayout.CENTER);

timer = new Timer(50, new TimerListener());

timer.setCoalesce(true);

timer.start();

}

public void actionPerformed(ActionEvent e) {

if (e.getSource() == startButton) {

timer.start();

} else if (e.getSource() == stopButton) {

timer.stop();

}

}

private class TimerListener implements ActionListener {

public void actionPerformed(ActionEvent e) {

wheel.repaint();

}

}

private class WheelComponent extends JComponent {

private static final int WHEEL\_SIZE = 200;

public void paintComponent(Graphics g) {

int x = getWidth() / 2 - WHEEL\_SIZE / 2;

int y = getHeight() / 2 - WHEEL\_SIZE / 2;

g.setColor(Color.LIGHT\_GRAY);

g.fillOval(x, y, WHEEL\_SIZE, WHEEL\_SIZE);

g.setColor(Color.BLACK);

g.drawOval(x, y, WHEEL\_SIZE, WHEEL\_SIZE);

int offset = (int) (System.currentTimeMillis() / 20) % 360;

g.setColor(Color.RED);

g.fillArc(x, y, WHEEL\_SIZE, WHEEL\_SIZE, offset, 30);

}

}

}

**Programme 5**

**write a Java Program that Accepts String Datafrom User and then Provide options forChanging case into Any of the Following. (UPPERCASE, lowercase,Sentence case. tOGGLE cASE)**

import java.util.Scanner;

public class StringCaseChanger {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

System.out.print("Enter a string: ");

String original = input.nextLine();

System.out.println("What do you want to change the case to?");

System.out.println("1. Uppercase");

System.out.println("2. Lowercase");

System.out.println("3. Sentence case");

System.out.println("4. Toggle case");

System.out.print("Enter your choice: ");

int choice = input.nextInt();

switch (choice) {

case 1:

System.out.println("Uppercase: " + original.toUpperCase());

break;

case 2:

System.out.println("Lowercase: " + original.toLowerCase());

break;

case 3:

System.out.println("Sentence case: " + toSentenceCase(original));

break;

case 4:

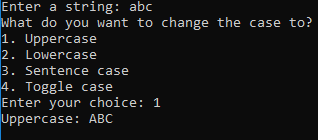
System.out.println("Toggle case: " + toggleCase(original));

break;

default:

System.out.println("Invalid choice.");

}

 }

public static String toSentenceCase(String str) {

StringBuilder result = new StringBuilder();

boolean capitalizeNext = true;

for (int i = 0; i < str.length(); i++) {

char currentChar = str.charAt(i);

if (currentChar == ' ' || currentChar == '.') {

capitalizeNext = true;

} else if (capitalizeNext) {

result.append(Character.toUpperCase(currentChar));

capitalizeNext = false;

} else {

result.append(Character.toLowerCase(currentChar));

}

}

return result.toString();

}

public static String toggleCase(String str) {

StringBuilder result = new StringBuilder();

for (int i = 0; i < str.length(); i++) {

char currentChar = str.charAt(i);

if (Character.isUpperCase(currentChar)) {

result.append(Character.toLowerCase(currentChar));

} else if (Character.isLowerCase(currentChar)) {

result.append(Character.toUpperCase(currentChar));

} else {

result.append(currentChar);

}

}

return result.toString();

}

}

**Programme 6:**

**Write an applet that simulates a Digital Clock displaying in the form Hour:Min: Sec. User should be able to change colour of Fonts of Clock by selection provided.**

/\*<applet code="digitalClock" width=200 height=60></applet>\*/

importjava.applet.\*;

importjava.awt.\*;

importjava.util.\*;

importjava.text.\*;

public class digitalClock extends Applet implements Runnable {

Thread t = null;

int hours=0, minutes=0, seconds=0;

String timeString = "";



public void init() {

setBackground(Color.green);

}

public void start() {

t = new Thread( this );

t.start();

}

public void run() {

try {

while (true) {

Calendar cal = Calendar.getInstance();

hours = cal.get( Calendar.HOUR\_OF\_DAY );

if ( hours > 12 ) hours -= 12;

minutes = cal.get( Calendar.MINUTE );

seconds = cal.get( Calendar.SECOND );

SimpleDateFormat formatter = new SimpleDateFormat("hh:mm:ss");

Date date = cal.getTime();

timeString = formatter.format( date );

repaint();

t.sleep( 1000 ); // interval given in milliseconds

}

}

catch (Exception e) { }

}

public void paint( Graphics g ) {

g.setColor(Color.blue );

g.drawString(timeString, 50, 50 );

}

}

**Programme 7:**

**Write a program to draw face Smiley with color using applet.**

importjava.awt.\*;

importjava.applet.\*;

/\*<applet code="cartoon" width="300" height="100"></applet>\*/

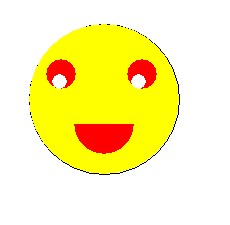
public class cartoon extends Applet

{

public void paint(Graphics g)

{

g.drawOval(40,40,150,150);//head

g.setColor(Color.yellow);

g.fillOval(40,40,150,150);

g.drawOval(62,75,30,30); //eye

g.drawOval(140,75,30,30); //eye

g.setColor(Color.red);

g.fillOval(57,75,30,30);

g.fillOval(138,75,30,30);

g.setColor(Color.white); //eye 2

g.fillOval(63,90,15,15);

g.fillOval(142,90,15,15);

g.setColor(Color.red); //teeth

g.fillArc(85,110,60,60,180,180);

}

}

**Programmme 8:**

**Write a program that accept Book information likeTitle, Author, Publication and Price for the N book fromthe user and display books in descending order withinterval of 1 second using thread.**

import java.util.\*;

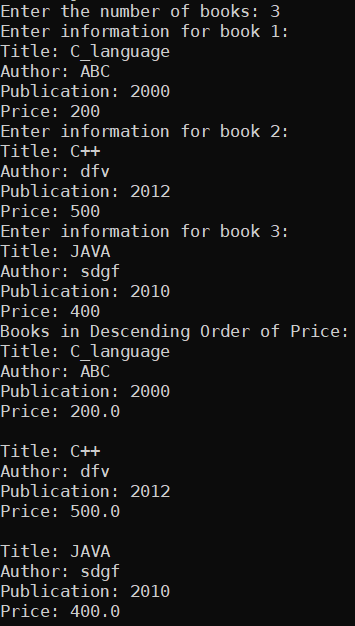
class Book {

String title;

String author;

String publication;

float price;

 public Book(String title, String author, String publication, float price) {

this.title = title;

this.author = author;

this.publication = publication;

this.price = price;

}

}

class BookDisplayThread extends Thread {

Book books[];

public BookDisplayThread(Book books[]) {

this.books = books;

}

public void run() {

System.out.println("Books in Descending Order of Price:");

for (int i = 0; i < books.length; i++) {

try {

Thread.sleep(1000);

} catch (InterruptedException e) {

System.out.println(e);

}

System.out.println(

"Title: " + books[i].title + "\nAuthor: " + books[i].author + "\nPublication: " + books[i].publication + "\nPrice: " + books[i].price + "\n");

}

}

}

public class BookInformation {

public static void main(String args[]) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the number of books: ");

int n = sc.nextInt();

Book books[] = new Book[n];

for (int i = 0; i < n; i++) {

System.out.println("Enter information for book " + (i + 1) + ":");

System.out.print("Title: ");

String title = sc.next();

System.out.print("Author: ");

String author = sc.next();

System.out.print("Publication: ");

String publication = sc.next();

System.out.print("Price: ");

float price = sc.nextFloat();

books[i] = new Book(title, author, publication, price);

}

/\*Arrays.sort(books, (a, b) -> Float.compare(b.price, a.price));\*/

BookDisplayThread thread = new BookDisplayThread(books);

thread.start();

}}

**Programme 9:**

**Writer a program to demonstrate general operations of circular link list using switch case.**

import java.util.\*;

class Node {

int data;

Node next;

public Node(int data) {

this.data = data;

this.next = null;

}

}

public class CircularLinkedList {

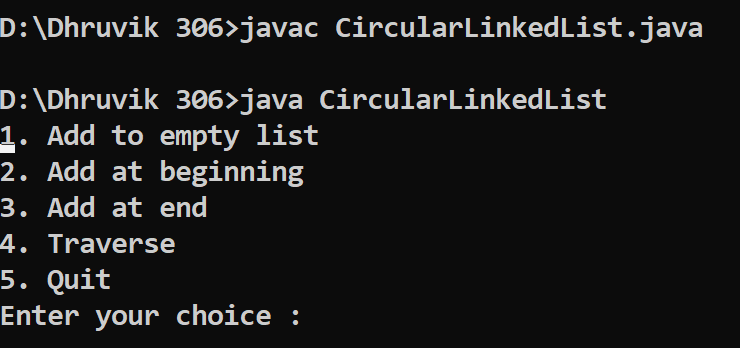
static Node last;

public static void addToEmpty(int data) {

if (last != null) {

System.out.println("List is not empty");

return;

 }

Node temp = new Node(data);

last = temp;

last.next = last;

}

public static void addBegin(int data) {

if (last == null) {

addToEmpty(data);

return;

}

Node temp = new Node(data);

temp.next = last.next;

last.next = temp;

}

public static void addEnd(int data) {

if (last == null) {

addToEmpty(data);

return;

}

Node temp = new Node(data);

temp.next = last.next;

last.next = temp;

last = temp;

}

public static void traverse() {

if (last == null) {

System.out.println("List is empty");

return;

}

Node temp = last.next;

while (temp != last) {

System.out.print(temp.data + " ");

temp = temp.next;

}

System.out.println(temp.data);

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

while (true) {

System.out.println("1. Add to empty list");

System.out.println("2. Add at beginning");

System.out.println("3. Add at end");

System.out.println("4. Traverse");

System.out.println("5. Quit");

System.out.print("Enter your choice : ");

int choice = sc.nextInt();

switch (choice) {

case 1:

System.out.print("Enter integer element to insert : ");

int data = sc.nextInt();

addToEmpty(data);

break;

case 2:

System.out.print("Enter integer element to insert : ");

data = sc.nextInt();

addBegin(data);

break;

case 3:

System.out.print("Enter integer element to insert : ");

data = sc.nextInt();

addEnd(data);

break;

case 4:

traverse();

break;

case 5:

System.exit(0);

default:

System.out.println("Wrong choice");

}

System.out.println();

}

}

}

**Programmme 10:**

**Writer a program to demonstrate general operations of singly link list using switch case.**

import java.util.\*;

class Node {

int data;

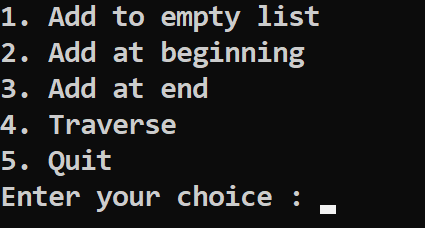
Node next;

public Node(int data) {

this.data = data;

this.next = null;

}

}

class LinkedList {

Node head;

public LinkedList() {

head = null;

}

public void insertAtStart(int data) {

Node newNode = new Node(data);

newNode.next = head;

head = newNode;

}

public void insertAtEnd(int data) {

Node newNode = new Node(data);

if (head == null) {

head = newNode;

return;

}

Node last = head;

while (last.next != null) {

last = last.next;

}

last.next = newNode;

}

public void insertAtPosition(int data, int position) {

Node newNode = new Node(data);

if (head == null) {

head = newNode;

return;

}

Node temp = head;

for (int i = 0; i < position - 1 && temp.next != null; i++) {

temp = temp.next;

}

newNode.next = temp.next;

temp.next = newNode;

}

Public void deleteAtStart() {

if (head == null) {

System.out.println("List is Empty");

return;

}

head = head.next;

}

public void deleteAtEnd() {

if (head == null) {

System.out.println("List is Empty");

return;

}

if (head.next == null) {

head = null;

return;

}

Node last = head;

Node prevToLast = null;

while (last.next != null) {

prevToLast = last;

last = last.next;

}

prevToLast.next = null;

}

public void deleteAtPosition(int position) {

if (head == null) {

System.out.println("List is Empty");

return;

}

if (position == 0) {

head = head.next;

return;

}

Node temp = head;

for (int i = 0; i < position - 1 && temp.next != null; i++) {

temp = temp.next;

}

if (temp.next == null) {

System.out.println("Position is out of the list");

return;

}

temp.next = temp.next.next;

}

public void displayList() {

if (head == null) {

System.out.println("List is Empty");

return;

}

Node temp = head;

while (temp != null) {

System.out.print(temp.data + " ");

temp = temp.next;

}

System.out.println();}}

**Programmme 11:**

**Write a program which demonstrate sunset using an Applet.**

/\* <applet code="TriangleInCircle" height=500 width=700></applet> \*/

import java.applet.Applet;

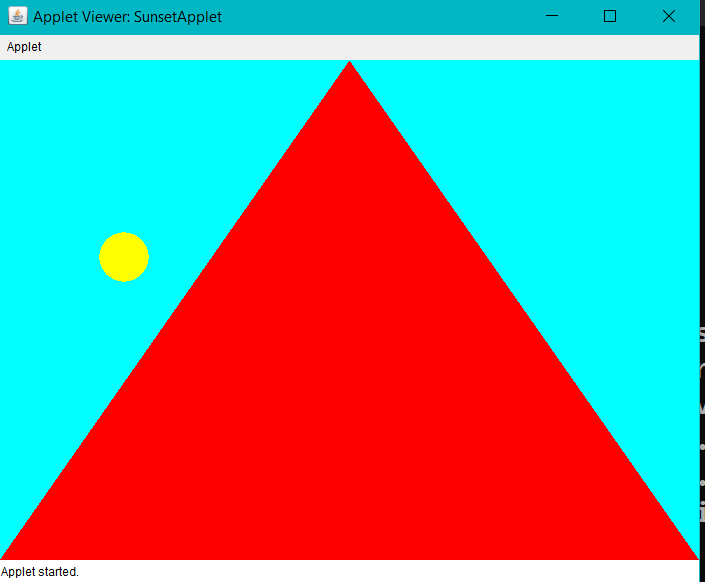
import java.awt.\*;

import java.util.Timer;

import java.util.TimerTask;

public class AnimatedSunsetApplet extends Applet {

private int sunRadius = 50;

 private int sunX = 100;

private int sunY = 100;

private int sunSpeed = 1;

private int skyY = 0;

private int skySpeed = 1;

public void init() {

setBackground(Color.CYAN);

Timer timer = new Timer();

timer.schedule(new TimerTask() {

public void run() {

sunY += sunSpeed;

skyY += skySpeed;

if (sunY + sunRadius >= getSize().height || sunY <= 0) {

sunSpeed = -sunSpeed;

}

if (skyY >= getSize().height) {

skyY = getSize().height;

}

repaint();

}

}, 0, 50);

}

public void paint(Graphics g) {

g.setColor(Color.YELLOW);

g.fillOval(sunX, sunY, sunRadius, sunRadius);

g.setColor(Color.RED);

int[] xPoints = {0, getSize().width, getSize().width / 2};

int[] yPoints = {skyY, skyY, 0};

g.fillPolygon(xPoints, yPoints, 3);

}}

**Programme 12:**

**Write a program add/sub/mul/div of 2 number in applet.**

/\*<applet code="q2" width=200 height=60></applet>\*/

importjava.awt.\*;

importjava.awt.event.\*;

importjava.applet.\*;

public class q2 extends Applet implements ActionListener

{

TextField t1 = new TextField(10);

TextField t2 = new TextField(10);

TextField t3 = new TextField(10);

Label l1 = new Label("FIRST NO=:");

Label l2 = new Label("SECOND NO:");



Label l4 = new Label("sub:");

Button b = new Button("ADD");

 Button c = new Button("sub");

Button d= new Button("mul");

Button e= new Button("divison");

public void init()

{

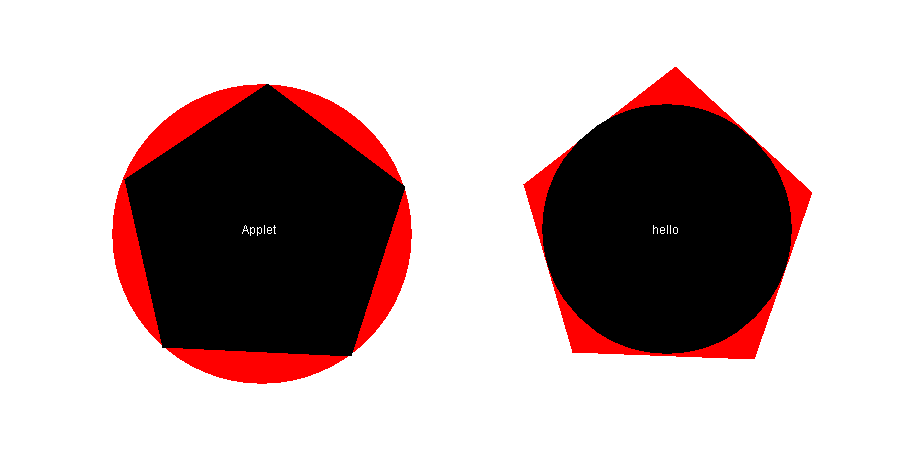
t1.setForeground(Color.red);

add(l1);

add(t1);

add(l2);

add(t2);

add(l4);

add(t3);

add(b);

add(c);

add(d);

add(e);

b.addActionListener(this);

c.addActionListener(this);

d.addActionListener(this);

e.addActionListener(this);

}

public void actionPerformed(ActionEvent e)

{

if (e.getSource() == b)

{

int n1 = Integer.parseInt(t1.getText());

int n2 = Integer.parseInt(t2.getText());

t3.setText(" " + (n1 + n2));

}

if (e.getSource() == c)

{

int n1 = Integer.parseInt(t1.getText());

int n2 = Integer.parseInt(t2.getText());

t3.setText(" " + (n1 - n2));

}

if (e.getSource() == d)

{

int n1 = Integer.parseInt(t1.getText());

int n2 = Integer.parseInt(t2.getText());

t3.setText(" " + (n1 \* n2));

}

if (e.getSource() == d)

{

int n1 = Integer.parseInt(t1.getText());

int n2 = Integer.parseInt(t2.getText());

t3.setText(" " + (n1 / n2));

}

}

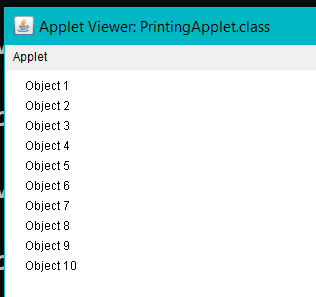
}

**Programmme 13:**

**Accept N number from applet tag and print that many numbers of object using an Applet.**

/\*<applet code="PrintingApplet.class" width="500" height="500">

<param name="numberOfObjects" value="10">

</applet>\*/

import java.applet.Applet;

import java.awt.\*;

public class PrintingApplet extends Applet {

private int numberOfObjects = 0;

public void init() {

String numberOfObjectsStr = getParameter("numberOfObjects");

if (numberOfObjectsStr != null) {

numberOfObjects = Integer.parseInt(numberOfObjectsStr);

}

}

public void paint(Graphics g) {

for (int i = 0; i < numberOfObjects; i++) {

g.drawString("Object " + (i + 1), 20, 20 \* (i + 1));

}

}

}

**Programmme 14:**

**Write a program which demonstrate functionality of creation and use of package.**

// File: com/example/util/MathUtil.java

package com.example.util;

public class MathUtil {

public static int add(int a, int b) {

return a + b;

}

public static int subtract(int a, int b) {

return a - b;

}

}

// File: com/example/Main.java

package com.example;

import com.example.util.MathUtil;

public class Main 007B

public static void main(String[] args) {

int a = 10;

int b = 20;

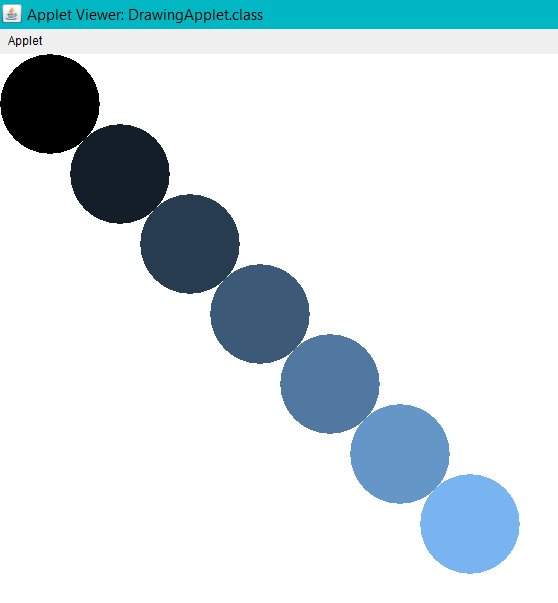
System.out.println("The sum of " + a + " and " + b + " is: " + MathUtil.add(a, b));

System.out.println("The difference of " + a + " and " + b + " is: " + MathUtil.subtract(a, b));

} }



**Programmme 15:**

**Accept number from applet tag parameter and draw that many numbers of objects. Fill them with different color.**

/\*<applet code="DrawingApplet.class" width="500" height="500">

<param name="numberOfShapes" value0">

</applet>\*/

import java.applet.Applet;

import java.awt.\*;

public class DrawingApplet extends Applet {

private int numberOfShapes = 0;

public void init() {

String numberOfShapesStr = getParameter("numberOfShapes");

if (numberOfShapesStr != null) {

numberOfShapes = Integer.parseInt(numberOfShapesStr);

}

}

public void paint(Graphics g) {

for (int i = 0; i < numberOfShapes; i++) {

g.setColor(new Color(i \* 20, i \* 30, i \* 40));

g.fillOval(i \* 70, i \* 70, 100, 100);

}

} }

**Programme16:**

importjava.applet.\*;

public class circleandtrangle extends Applet

{

int x1[]={405,262,300,489,543};

int y1[]={149,245,413,422,252};

int n1=5;

int x2[]={813,661,710,892,950};

int y2[]={132,250,418,425,258};

int n2=5;

public void paint (Graphics g)

{

g.setColor(Color.red);

g.fillArc(250,150,300,300,0,360);

g.setColor(Color.black);

g.fillPolygon(x1,y1,n1);

g.setColor(Color.white);

g.drawString("Applet",380,300);//

g.setColor(Color.red);

g.fillPolygon(x2,y2,n2);

g.setColor(Color.black);

g.fillArc(680,170,250,250,0,360);

g.setColor(Color.white);

g.drawString("hello",790,300);} }

