(Practical 1)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 1:

// Design an application to demonstrate the Radio Button and Check box.

import java.awt.\*;

class PracticalNo\_1Q1 extends Frame{

public PracticalNo\_1Q1(){

Frame myFrame = new Frame("Radion Button & Checkbox");

myFrame.setLayout(new FlowLayout());

myFrame.setTitle("Radio & Checkbox Button");

myFrame.setSize(500, 500);

myFrame.setVisible(true);

myFrame.setResizable(true);

Label infoOfProgrammer = new Label("// Programmer: Harsh Kale");

Font myFont = new Font("Lucida Console", Font.BOLD, 17);

Font myNewFont = new Font("Arial", Font.BOLD, 17);

Label myLable = new Label("This is a program Demonstrating the Radion button and Check box!");

myLable.setFont(myFont);

Checkbox myCheckboxOne = new Checkbox("Python", true);

myCheckboxOne.setFont(myNewFont);

Checkbox myCheckboxTwo = new Checkbox("JavaScript", true);

myCheckboxTwo.setFont(myNewFont);

CheckboxGroup myCheckboxGroup = new CheckboxGroup();

Checkbox radioBtnOne = new Checkbox("Genuis Programmer", true, myCheckboxGroup);

radioBtnOne.setFont(myNewFont);

Checkbox radioBtnTwo = new Checkbox("Hello world Programmer", true, myCheckboxGroup);

radioBtnTwo.setFont(myNewFont); Component[] myObjects = {infoOfProgrammer, myLable, myCheckboxOne, myCheckboxTwo, radioBtnOne, radioBtnTwo};

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

myFrame.add(myObjects[i]);

}

}

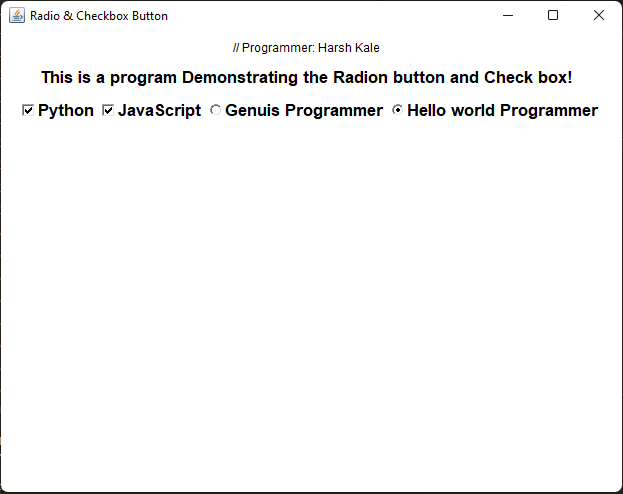
public static void main(String[] args) {

new PracticalNo\_1Q1();

}

}

Output:



(Practical 1)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 2:

// Design an application to create a form with the use of text field, text area, button and label.

import java.awt.\*;

public class PracticalNo\_1Q2 extends Frame {

public PracticalNo\_1Q2(){

setLayout(new FlowLayout());

setTitle("Application of Text Field, Text Area, Button and Label!");

setSize(700, 700);

setVisible(true);

Font ff1 = new Font("Times New Roman", Font.BOLD, 12);

Font ff2 = new Font("Arial", Font.ITALIC, 17);

Label label1 = new Label("Application of Textfield, Textarea, Button and Label! By Developer Harsh Kale");

label1.setFont(ff1);

label1.setBounds(15, 40, 450, 30);

Label label2 = new Label("Form", Label.CENTER);

label2.setFont(ff2);

label2.setBounds(210, 90, 80, 20);

Label ln = new Label("Enter Your Name: ", Label.LEFT);

TextField tfn = new TextField();

ln.setBounds(30, 140, 110, 20);

tfn.setBounds(180, 140, 250, 20);

Label rollno = new Label("Enter Your Roll No.: ", Label.LEFT);

TextField tfrn = new TextField();

rollno.setBounds(30, 180, 150, 20);

tfrn.setBounds(180, 180, 250, 20);

Label addrs = new Label("Enter Your Address: ", Label.LEFT);

TextArea taddress = new TextArea();

addrs.setBounds(30, 220, 170, 20);

taddress.setBounds(180, 250, 250, 125);

Button submit = new Button("Submit!");

Button reset = new Button("Reset!!");

submit.setBounds(200, 425, 100, 30);

reset.setBounds(200, 455, 100, 30);

add(label1);

add(label2);

add(ln);

add(rollno);

add(tfn);

add(tfrn);

add(addrs);

add(taddress);

add(submit);

add(reset);

}

public static void main(String[] args) {

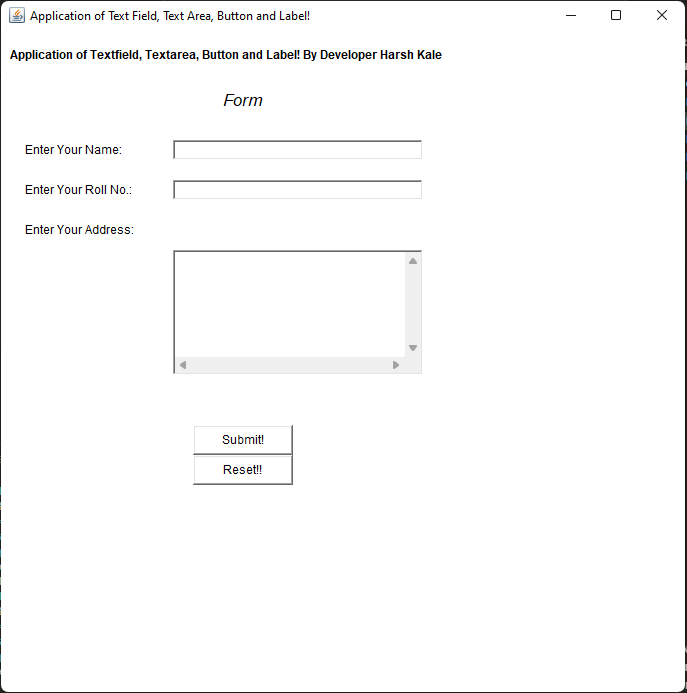
System.out.println("Hello, world Programmer! Harsh Moreshwar Kale");

new PracticalNo\_1Q2();

}

}

Output:



(Practical 1)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 3:

// Develop a program using Label to display the message "Welcome to Java Programming".

import java.awt.\*;

public class PracticalNo\_1Q3 extends Frame {

public PracticalNo\_1Q3(){

setTitle("Displaying the \'Welcome to Java Programming\' message on the frame!");

setSize(700, 700);

setVisible(true);

Font ff = new Font("Arial", Font.ITALIC, 20);

Label l = new Label("Welcome to Java Programming", Label.LEFT);

l.setFont(ff);

add(l);

}

public static void main(String[] args) {

System.out.println("Developer Harsh Moreshwar Kale!");

new PracticalNo\_1Q3();

}

}

Output:



(Practical 1)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 4:

// Develop a Program to Select Multiple Languages known to User.

import java.awt.\*;

public class PracticalNo\_1Q4 extends Frame{

public PracticalNo\_1Q4(){

setLayout(new FlowLayout());

setTitle("Advance Java Program!");

setSize(700, 700); setVisible(true);

Label l = new Label("Select from the following! which language you used in your daily life!");

Checkbox cmr = new Checkbox("Marathi (मराठी)");

Checkbox chi = new Checkbox("Hindi");

Checkbox csk = new Checkbox("Sanskrit");

Checkbox cpy = new Checkbox("Python");

Checkbox cc = new Checkbox("C");

add(l); add(cmr); add(chi); add(csk); add(cpy); add(cc);

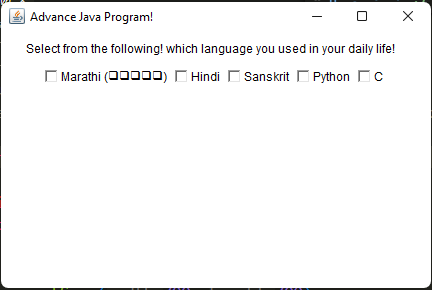
}

public static void main(String[] args) {

System.out.println("Developer Harsh Moreshwar Kale");

new PracticalNo\_1Q4(); }}

Output:



(Practical 1)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 5:

// Develop a Program to Create 3 buttons with caption ok, reset, cancel.

import java.awt.\*;

public class PracticalNo\_1Q5 extends Frame{

public PracticalNo\_1Q5(){

setLayout(new FlowLayout());

setTitle("Advance Java Programming By Harsh Kale!");

setSize(700, 700);

setVisible(true);

Label l = new Label("Developer Harsh Moreshwar Kale, Click through the following buttons!");

Button ok = new Button("OK");

Button reset = new Button("RESET");

Button cancel = new Button("CANCEL");

add(l); add(ok); add(reset); add(cancel);

}

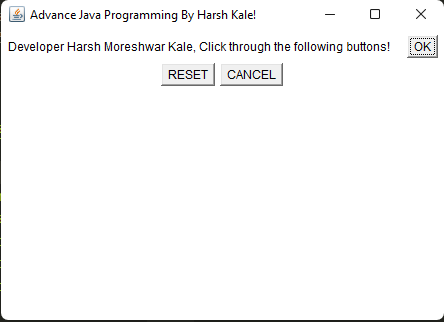
public static void main(String[] args) {

System.out.println("Developer Harsh Moreshwar Kale!");

new PracticalNo\_1Q5();

}}

Output:



(Practical 2)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 1:

// Write a java program to show following output in list!

import java.awt.\*;

public class PracticalNo\_2Q1 extends Frame {

public PracticalNo\_2Q1(){

setLayout(new FlowLayout());

setTitle("Demo of List in Advance Java Programming!");

setSize(700, 700);

setVisible(true);

Label l = new Label("Here is the program of list in advance java programming by Harsh Kale!");

List list = new List(3, false);

list.add("Summer");

list.add("Winter");

list.add("Rainy"); add(l); add(list);

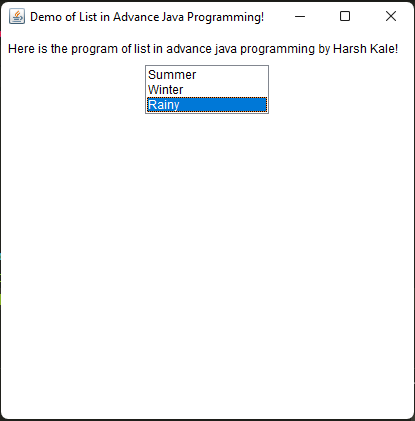
}

public static void main(String[] args) {

System.out.println("Developer Harsh Moreshwar Kale!");

new PracticalNo\_2Q1(); }}

Output:



(Practical 2)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 2:

// Develop an application using list components to add names of 10 different cities.abstract

import java.awt.\*;

public class PracticalNo\_2Q2 extends Frame {

public PracticalNo\_2Q2(){

setLayout(new FlowLayout());

setTitle("List Components In Advance Java Programming");

setSize(700, 700);

setVisible(true);

Label l = new Label("This Program is created by Harsh Kale!");

List list = new List(4, false);

list.add("Latur"); list.add("Barshi"); list.add("Solapur"); list.add("Nanded");

add(l);

add(list);

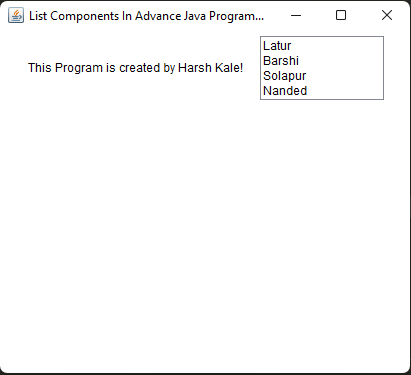
}

public static void main(String[] args) {

System.out.println("Developer Harsh Moreshwar Kale");

new PracticalNo\_2Q2(); } }

Output:



(Practical 2)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 3:

// Develop an application select multiple names of news paper.

import java.awt.\*;

public class PracticalNo\_2Q3 extends Frame {

public PracticalNo\_2Q3(){

setLayout(new FlowLayout());

setTitle("Program of List Components in Advance Java Programming!");

setSize(700, 700);

setVisible(true);

Label l = new Label("This is a program of compoenents to select the names of news papers by Harsh Kale!");

List list = new List(4, true);

list.add("Lokmat");

list.add("The New Indian Times");

list.add("The Hindu"); list.add("Dyandeep"); list.add("Maradhi Paper"); add(l); add(list);

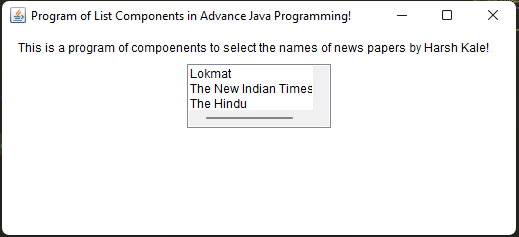
}

public static void main(String[] args) {

System.out.println("Developer Harsh Moreshwar Kale");

new PracticalNo\_2Q3(); }}

Output:



(Practical 3)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 1:

// Write a java program to demonstrate the use of grid layout of 5 \* 5

import java.awt.\*;

public class PracticalNo\_3Q1 extends Frame {

PracticalNo\_3Q1(){

setLayout(new GridLayout(5, 5));

setTitle("Program of Grid Layout In Advance Java Programin by Harsh kale");

setSize(700, 700);

setVisible(true);

for(int i = 1; i <= 20; i++){

add(new Label("Cell " + i));

}

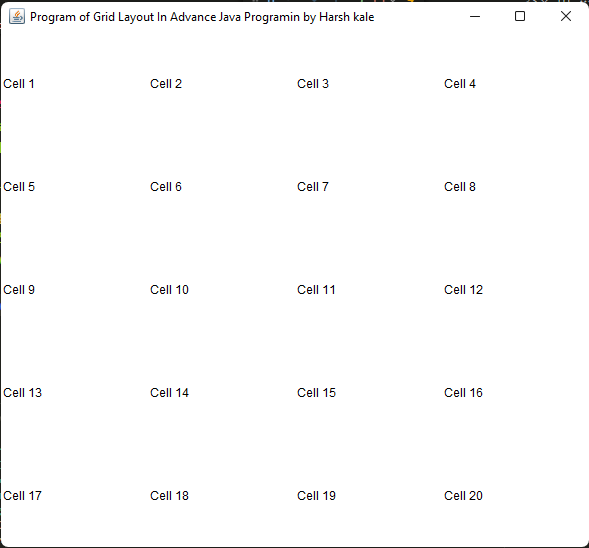
}

public static void main(String[] args) {

System.out.println("Developer Harsh Moreshwar Kale");

new PracticalNo\_3Q1(); }}

Output:



(Practical 3)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 2:

// Write a java program to display the No. of buttons from 0 to 0.

import java.awt.\*;

public class PracticalNo\_3Q2 extends Frame{

PracticalNo\_3Q2(){

setLayout(new GridLayout(3, 3));

setTitle("Program of Grid layout in Advance Java Programing by Harsh Kale!");

setSize(700, 700); setVisible(true);

for(int i = 0; i <= 9; i++){

add(new Button("Harsh " + i));

}

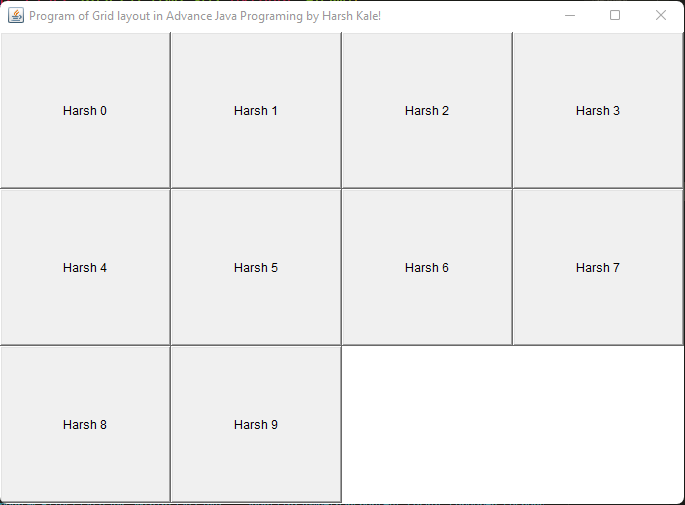
}

public static void main(String[] args) {

System.out.println("Developer Harsh Moreshwar Kale!");

new PracticalNo\_3Q2(); }}

Output:



(Practical 3)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 3:

// Write a java program to display the No. of buttons from 0 to 0.

import java.awt.\*;

public class PracticalNo\_3Q3 extends Frame{

PracticalNo\_3Q3(){

setLayout(new GridLayout(3, 2, 20, 20));

setTitle("Program of Grid Layout in Advance Java Programing by Harsh Kale!");

setSize(700, 700);

setVisible(true);

for(int i = 0; i <= 7; i++){

Button btn = new Button("Harsh " + i); add(btn);

}

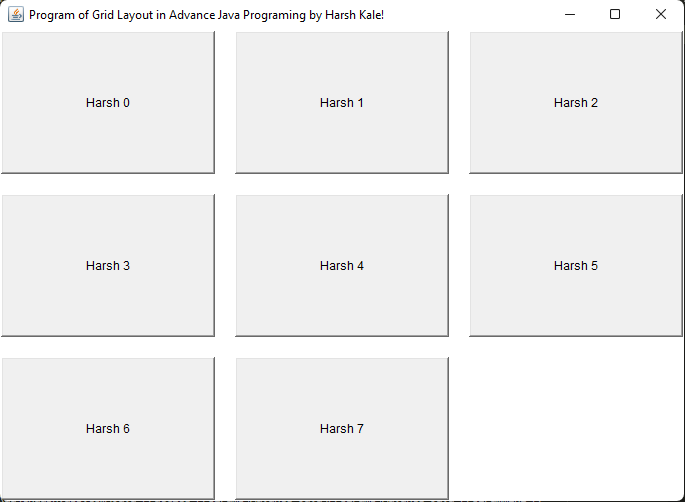
}

public static void main(String[] args) {

System.out.println("Developer Harsh Moreshwar Kale!");

new PracticalNo\_3Q3(); } }

Output:



(Practical 3)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 4: // Write a java program to display the use of border layout.!

import java.awt.\*;

public class PracticalNo\_3Q4 extends Frame{

PracticalNo\_3Q4(){

setTitle("Program of border layout in advance java programming");

setSize(700, 700); setVisible(true);

setLayout(new BorderLayout(10, 10));

Button northButton = new Button("North"); add(northButton, BorderLayout.NORTH);

Button southButton = new Button("South"); add(southButton, BorderLayout.SOUTH);

Button eastButton = new Button("East"); add(eastButton, BorderLayout.EAST);

Button wesButton = new Button("West"); add(wesButton, BorderLayout.WEST);

Button centerButton = new Button("Center"); add(centerButton, BorderLayout.CENTER);

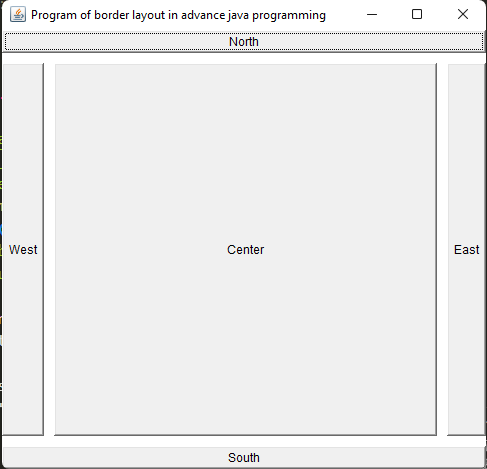
}

public static void main(String[] args) {

System.out.println("Developer Harsh Moreshwar Kale!");

new PracticalNo\_3Q4(); }}

Output:



(Practical 4)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 1:

// Write the Output of the following program!

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

public class PracticalNo\_4Q1 extends JFrame implements ActionListener{

CardLayout card; JButton btn1, btn2, btn3; Container c;

PracticalNo\_4Q1(){

c = getContentPane(); card = new CardLayout(40, 30); c.setLayout(card);

btn1 = new JButton("ReactJS"); btn2 = new JButton("NodeJS");

btn3 = new JButton("VueJS"); btn1.addActionListener(this);

btn2.addActionListener(this); btn3.addActionListener(this);

c.add("A", btn1); c.add("B", btn2); c.add("C", btn3);

}

public void actionPerformed(ActionEvent e){ card.next(c); }

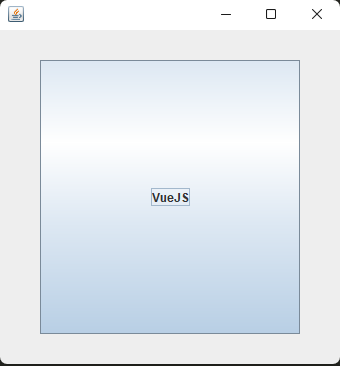
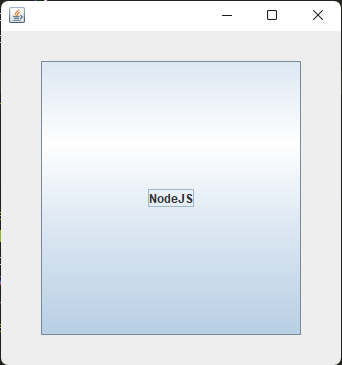
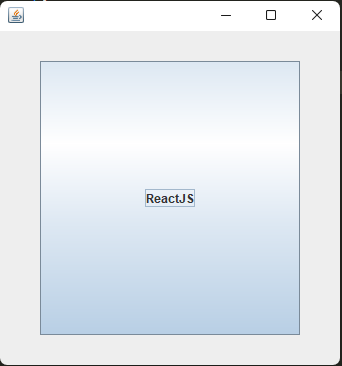
public static void main(String[] args) {

System.out.println("Developer Harsh Moreshwar Kale!");

PracticalNo\_4Q1 p = new PracticalNo\_4Q1();

p.setSize(700, 700); p.setVisible(true); p.setDefaultCloseOperation(EXIT\_ON\_CLOSE); }}

Output:



(Practical 4)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 2:

// Write a java program to display the output of the following code.

import java.awt.\*;

import javax.swing.\*;

public class PracticalNo\_4Q2 extends JFrame {

PracticalNo\_4Q2() {

Label l = new Label("GridBagLayout Program by Harsh Kale!");

add(l);

GridBagLayout grid = new GridBagLayout(); GridBagConstraints gbc = new GridBagConstraints();

setLayout(grid);

setTitle("GridBag Layout Program By Harsh Kale");

GridBagLayout layout = new GridBagLayout();

this.setLayout(layout);

gbc.fill = GridBagConstraints.HORIZONTAL;

gbc.gridx = 0;

gbc.gridy = 0;

this.add(new Button("Button One"), gbc);

gbc.gridx = 1;

gbc.gridy = 0;

this.add(new Button("Button two"), gbc);

gbc.fill = GridBagConstraints.HORIZONTAL;

gbc.ipady = 20;

gbc.gridx = 0;

gbc.gridy = 1;

this.add(new Button("Button Three"), gbc);

gbc.gridx = 1;

gbc.gridy = 1;

this.add(new Button("Button Four"), gbc);

gbc.gridx = 0;

gbc.gridy = 2;

gbc.fill = GridBagConstraints.HORIZONTAL;

gbc.gridwidth = 2;

this.add(new Button("Button Five"), gbc);

setSize(700, 700);

setPreferredSize(getSize());

setVisible(true);

setDefaultCloseOperation(EXIT\_ON\_CLOSE);

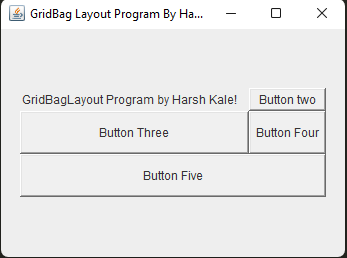
}

public static void main(String[] args) {

System.out.println("Developer Harsh MOreshwar Kale!");

new PracticalNo\_4Q2(); }}

Output:



(Practical 4)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 3:

// Write a java program to display following output of GridBagLayout.

import java.awt.\*;

import javax.swing.\*;

public class PracticalNo\_4Q3 extends JFrame {

PracticalNo\_4Q3(){

setSize(700, 700);

setPreferredSize(getSize());

setVisible(true);

setDefaultCloseOperation(EXIT\_ON\_CLOSE);

GridBagLayout grid = new GridBagLayout();

GridBagConstraints gbc = new GridBagConstraints();

setLayout(grid);

setTitle("GridBag Layout Example By Harsh Kale");

gbc.fill = GridBagConstraints.HORIZONTAL;

gbc.gridx = 0; gbc.gridy = 0; this.add(new Label("Name: "), gbc);

gbc.gridx = 1;

gbc.gridy = 0;

this.add(new TextField("Harsh", 1), gbc);

gbc.fill = GridBagConstraints.HORIZONTAL;

gbc.gridx = 0;

gbc.gridy = 1;

gbc.weightx=0;

gbc.weighty=0;

this.add(new Label("Message"), gbc);

gbc.gridx = 1; gbc.gridy = 1;

this.add(new TextArea(3, 5), gbc); gbc.gridx = 0; gbc.gridy = 2;

gbc.fill = GridBagConstraints.HORIZONTAL;

gbc.gridwidth = 2;

gbc.gridheight=1;

gbc.insets= new Insets (30, 0, 10, 0); this.add(new JButton("Submit"), gbc);

}

public static void main(String[] args) {

System.out.println("Developer Harsh Moreshwar Kale!");

new PracticalNo\_4Q3();

}

}

Output:



(Practical 5)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 1:

// Write a java program to create menu of different colors and disable menu item for Black Color.

import java.awt.\*;

public class PracticalNo\_5Q1 extends Frame{

MenuBar mb; Menu colorNameMenu;

MenuItem redItem, orangItem, blueItem, blackItem;

PracticalNo\_5Q1(){

setTitle("Advane Java Menu Program By Harsh Kale");

setSize(700, 700); mb = new MenuBar();

colorNameMenu = new Menu("Colors");

redItem = new MenuItem("Red");

orangItem = new MenuItem("Orange");

blueItem = new MenuItem("Blue");

blackItem = new MenuItem("Black");

blackItem.setEnabled(false); colorNameMenu.add(redItem);

colorNameMenu.add(orangItem); colorNameMenu.add(blueItem);

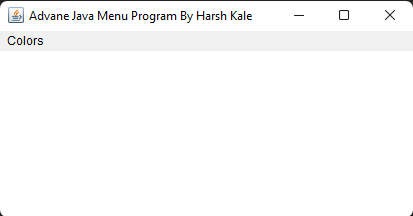
colorNameMenu.add(blackItem); mb.add(colorNameMenu);

setMenuBar(mb); setVisible(true);

}

public static void main(String[] args) { System.out.println("Developer Harsh Moreshwar Kale"); new PracticalNo\_5Q1(); }}

Output:



(Practical 5)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 2:

// Find an error and correct it also display the output after corrections.

import java.awt.\*;

import java.awt.event.KeyEvent;

public class PracticalNo\_5Q2 extends Frame{

MenuBar mb; MenuItem m1, m2, m3; Menu mn;

MenuShortcut ms;

PracticalNo\_5Q2(){

setTitle("Menubar Program By Harsh Kale!");

setSize(700, 700); setLayout(null);

ms = new MenuShortcut(KeyEvent.VK\_X);

mn = new Menu("File"); mb = new MenuBar();

m1 = new MenuItem("Open with VS Code");

m2 = new MenuItem("Auto Save");

m3 = new MenuItem("Harsh Kale");

mn.add(m1); mn.add(m2); mn.addSeparator(); mn.add(m3);

mb.add(mn); setMenuBar(mb); setVisible(true);

}

public static void main(String[] args) {

System.out.println("Developer Harsh Moreshwar Kale");

new PracticalNo\_5Q2(); }}

Output:



(Practical 6)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 1:

// Write a Program for Following output!

import java.awt.FlowLayout; import javax.swing.\*;

public class PracticalNo\_6Q1 extends JFrame {

PracticalNo\_6Q1(){

super("ComboBox Program by Harsh Kale"); setSize(700, 700);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

String cities[] = {"Solapur", "Barshi", "Latur", "Banglore"};

JComboBox<String> comboBox = new JComboBox<>(cities);

JScrollPane scrollPane = new JScrollPane(comboBox);

add(scrollPane); setVisible(true);

setLayout(new FlowLayout());

}

public static void main(String[] args) {

System.out.println("Developer Harsh Kale");

new PracticalNo\_6Q1(); }}

Output:



(Practical 6)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 2:

// Program using JComboBox to select different states of India or programming languages.

import java.awt.\*; import javax.swing.JComboBox; import javax.swing.JFrame;

import javax.swing.JScrollPane;

public class PracticalNo\_6Q2 extends JFrame{

PracticalNo\_6Q2(){

super("ComboBox Program by Harsh Kale"); setSize(700, 700);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

String cities[] = {"C", "C++", "C#", "Java", "Python", "R"};

JComboBox<String> comboBox = new JComboBox<>(cities);

JScrollPane scrollPane = new JScrollPane(comboBox);

add(scrollPane); setVisible(true); setLayout(new FlowLayout());

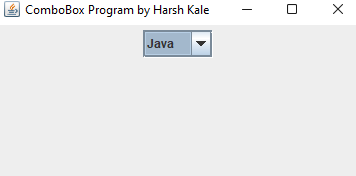
}

public static void main(String[] args) {

System.out.println("Developer Harsh Kale");

new PracticalNo\_6Q2(); }}

Output:



(Practical 6)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 3:

// Program using JScrollPane in Advance Java Programming.

import javax.swing.\*; import java.awt.\*;

public class PracticalNo\_6Q3 extends JFrame {

PracticalNo\_6Q3(){

super("ScrollPane Program in Advance Java Programming By Harsh Kale!");

setLayout(new BorderLayout()); setSize(400, 400);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

JTextArea t = new JTextArea();

for(int i = 0; i < 100; i++){ t.append("Hello, world programmer Harsh Kale \n"); }

JScrollPane scrollPane = new JScrollPane(t);

add(scrollPane, BorderLayout.CENTER); setVisible(true);

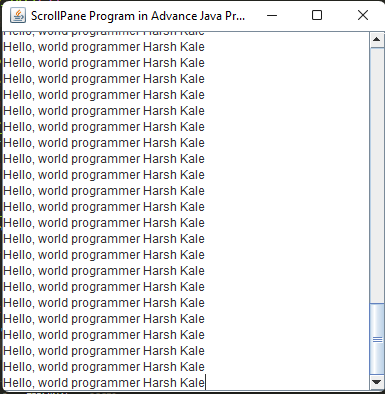
}

public static void main(String[] args) {

System.out.println("Developer Harsh Kale"); new PracticalNo\_6Q3();

}}

Output:



(Practical 7)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 1:

// Write a Program for JTree Component!

import javax.swing.\*; import javax.swing.tree.\*;

public class PracticalNo\_7Q1 extends JFrame{

PracticalNo\_7Q1(){

setTitle("JTree Program By Harsh Kale!"); setVisible(true);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

DefaultMutableTreeNode root = new DefaultMutableTreeNode("root");

DefaultMutableTreeNode n1 = new DefaultMutableTreeNode("Node first");

DefaultMutableTreeNode n2 = new DefaultMutableTreeNode("Node second");

DefaultMutableTreeNode n3 = new DefaultMutableTreeNode("Node third");

DefaultMutableTreeNode n4 = new DefaultMutableTreeNode("Node fourth");

n1.add(n3); n1.add(n4);

root.add(n1); root.add(n2);

JTree tree = new JTree(root); JScrollPane scrollPane = new JScrollPane(tree);

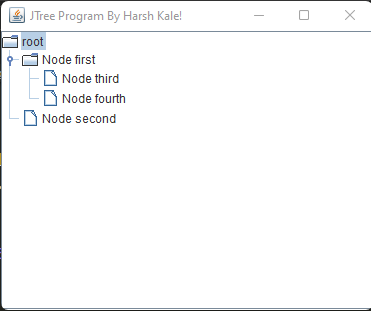
getContentPane().add(scrollPane); pack(); setSize(700, 700);

}

public static void main(String[] args) {

System.out.println("Developer Harsh Moreshwar Kale"); new PracticalNo\_7Q1(); }}

Output:



(Practical 7)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 1: Write a Program for Following output.

import javax.swing.\*; import javax.swing.tree.\*;

public class PracticalNo\_7Q2 extends JFrame {

PracticalNo\_7Q2() { setTitle("JTree Program By Harsh Kale"); setVisible(true);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

DefaultMutableTreeNode rootNode = new DefaultMutableTreeNode("India");

DefaultMutableTreeNode node1 = new DefaultMutableTreeNode("Maharashtra");

DefaultMutableTreeNode node2 = new DefaultMutableTreeNode("Gujrat");

DefaultMutableTreeNode node11 = new DefaultMutableTreeNode("Mumbai");

DefaultMutableTreeNode node12 = new DefaultMutableTreeNode("Pune");

DefaultMutableTreeNode node13 = new DefaultMutableTreeNode("Nashik");

DefaultMutableTreeNode node14 = new DefaultMutableTreeNode("Nagpur");

DefaultMutableTreeNode node15 = new DefaultMutableTreeNode("Latur");

node1.add(node11); node1.add(node12); node1.add(node13); node1.add(node14);

node1.add(node15); rootNode.add(node1); rootNode.add(node2);

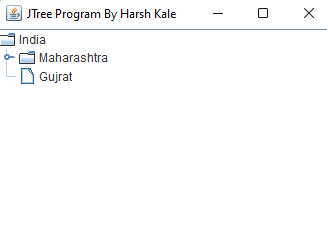
JTree tree = new JTree(rootNode); JScrollPane scrollPane = new JScrollPane(tree);

getContentPane().add(scrollPane); pack();

}

public static void main(String[] args) { System.out.println("Developer Harsh Kale!"); new PracticalNo\_7Q2(); }}

Output:



(Practical 7)

// Programmer: Harsh Moreshwar Kale Created Date: 13/09/2023

// Question 3: Program to show root directory and its sub folder of your system.

import java.awt.\*; import java.io.\*; import javax.swing.\*; import javax.swing.tree.\*;

public class PracticalNo\_7Q3 {

public static void main(String[] args) {

SwingUtilities.invokeLater(() -> {

JFrame frame = new JFrame("JTree Program");

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

File rootDir = new File("D:/");

DefaultMutableTreeNode root = new DefaultMutableTreeNode(rootDir);

DefaultTreeModel treeModel = new DefaultTreeModel(root); addSubfolders(root, rootDir);

JTree tree = new JTree(treeModel);

tree.setPreferredSize(new Dimension(300, 200));

frame.add(tree, BorderLayout.CENTER);

frame.pack(); // Use pack() to set the frame size based on its contents

frame.setVisible(true);

}); }

private static void addSubfolders(DefaultMutableTreeNode parent, File dir) {

File[] subDirs = dir.listFiles();

if (subDirs != null) {

for (File subDir : subDirs) {

if (subDir.isDirectory()) {

DefaultMutableTreeNode child = new DefaultMutableTreeNode(subDir);

parent.add(child);

addSubfolders(child, subDir);

} } } }}

(Practical 8)

// Programmer: Harsh Moreshwar Kale Created Date: 13/09/2023

// Question 1: Develop a program to demonstrate the use of JTable.

import javax.swing.\*; import javax.swing.table.DefaultTableModel; import java.awt.\*;

public class PracticalNo\_8Q1 {

public static void main(String[] args) {

SwingUtilities.invokeLater(() -> {

JFrame frame = new JFrame("JTable Program By Harsh Kale");

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

String[] columnNames = {"Name", "Age", "Country"};

Object[][] data = {

{"Harsh", 70, "Latur"},

{"Sanket", 25, "Sambhaji-Nagar"},

{"Wadkar", 35, "Moti Nagar"},

{"Omkar", 28, "Chincholi"},

{"Kale", 40, "Barshi"} };

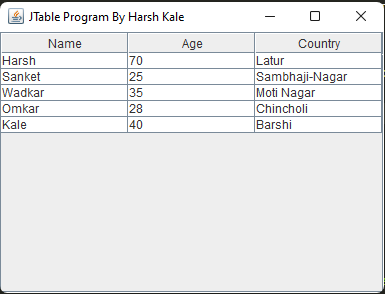
DefaultTableModel model = new DefaultTableModel(data, columnNames);

JTable table = new JTable(model); JScrollPane scrollPane = new JScrollPane(table);

frame.add(scrollPane, BorderLayout.CENTER);

frame.pack(); frame.setSize(400, 300); frame.setVisible(true); }); }}

Output:



(Practical 8)

// Programmer: Harsh Moreshwar Kale Created Date: 13/09/2023

// Question 2: Program to show root directory and its sub folder of your system.

import javax.swing.JFrame; import javax.swing.JScrollPane;

import javax.swing.JTable; import javax.swing.SwingUtilities;

import javax.swing.table.DefaultTableModel; import java.awt.\*;

public class PracticalNo\_8Q2 { public static void main(String[] args) {

System.out.println("Developer Harsh Moreshwar Kale");

SwingUtilities.invokeLater(() -> {

JFrame frame = new JFrame("JTable Program By Harsh Kale");

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

// Create sample data for the table

String[] columnNames = {"ID", "Name", "Salary"};

Object[][] data = {

{1, "Harsh Kale", "70,00,00,000"},

{2, "Omanand Swami", "1200"},

{3, "Prathmesh Bavge", "200"},

{4, "Akshay Gitte", "2700"},

{5, "Amay Devshatwar", "Infinity"} };

// Create a DefaultTableModel

DefaultTableModel model = new DefaultTableModel(data, columnNames);

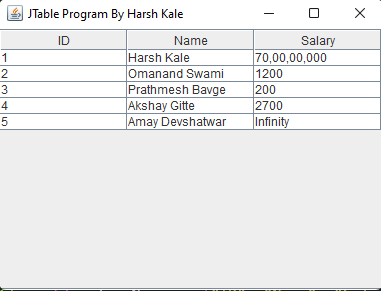
JTable table = new JTable(model); JScrollPane scrollPane = new JScrollPane(table);

frame.add(scrollPane, BorderLayout.CENTER); frame.pack();

frame.setSize(400, 300); frame.setVisible(true);

}); }}

Output:



(Practical 8)

// Programmer: Harsh Moreshwar Kale Created Date: 13/09/2023

// Question 3: Program to show table view of 10 students. (Name, Percentage, Grade).

import javax.swing.\*; import javax.swing.table.DefaultTableModel; import java.awt.\*;

public class PracticalNo\_8Q3 {

public static void main(String[] args) {

SwingUtilities.invokeLater(() -> {

JFrame frame = new JFrame("Student Table View");

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

// Define column names

String[] columnNames = {"Name", "Percentage", "Grade"};

// Create sample data for 10 students

Object[][] data = {

{"Student 1", 85.5, "A"}, {"Student 2", 78.0, "B"}, {"Student 3", 92.3, "A"},

{"Student 4", 63.7, "C"}, {"Student 5", 77.8, "B"}, {"Student 6", 88.2, "A"},

{"Student 7", 72.5, "B"}, {"Student 8", 95.1, "A"}, {"Student 9", 61.9, "C"},

{"Student 10", 84.6, "B"} };

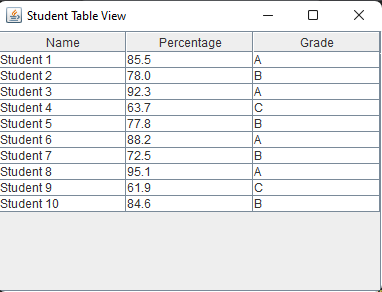
DefaultTableModel model = new DefaultTableModel(data, columnNames);

JTable table = new JTable(model); JScrollPane scrollPane = new JScrollPane(table);

frame.add(scrollPane, BorderLayout.CENTER); frame.pack();

frame.setSize(400, 300); frame.setVisible(true); }); }}

Output:



(Practical 9)

// Programmer: Harsh Moreshwar Kale Created Date: 13/09/2023

// Question 1: Write a program to launch a JProgressBar.

import javax.swing.\*; import java.awt.event.ActionEvent; import java.awt.event.ActionListener;

public class PracticalNo\_9Q1 { public static void main(String[] args) {

SwingUtilities.invokeLater(() -> {

JFrame frame = new JFrame("JProgressBar By Harsh Kale!");

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setSize(300, 100); JProgressBar progressBar = new JProgressBar(0, 100);

progressBar.setStringPainted(true); JButton startButton = new JButton("Start Progress");

startButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

Thread taskThread = new Thread(new Runnable() {

public void run() {

for (int i = 0; i <= 100; i++) {

final int progressValue = i;

SwingUtilities.invokeLater(new Runnable() {

public void run() {

progressBar.setValue(progressValue); } });

try {

Thread.sleep(100); // Simulate some work

} catch (InterruptedException ex) {

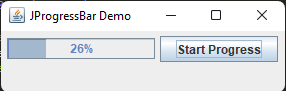
ex.printStackTrace(); } } }); taskThread.start(); } });

JPanel panel = new JPanel(); panel.add(progressBar);

panel.add(startButton); frame.add(panel); frame.setVisible(true);

}); }}

Output:



(Practical 9)

// Programmer: Harsh Moreshwar Kale

// Created Date: 13/09/2023

// Question 2:

// Develop a Program to Demonstrate the use of JProgressBar.

import javax.swing.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class PracticalNo\_9Q2 {

private static JProgressBar progressBar;

private static JButton startButton;

public static void main(String[] args) {

SwingUtilities.invokeLater(() -> {

JFrame frame = new JFrame("JProgressBar Program By Harsh Kale");

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setSize(300, 100);

progressBar = new JProgressBar(0, 100);

progressBar.setStringPainted(true);

startButton = new JButton("DownLoad MugBit!");

startButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

startProgress(); } });

JPanel panel = new JPanel();

panel.add(progressBar); panel.add(startButton);

frame.add(panel); frame.setVisible(true); }); }

private static void startProgress() {

startButton.setEnabled(false); // Disable the button while the task is running

SwingWorker<Void, Integer> worker = new SwingWorker<Void, Integer>() {

@Override protected Void doInBackground() throws Exception {

for (int i = 0; i <= 100; i++) {

Thread.sleep(100); // Simulate work (100 milliseconds) publish(i); // Publish progress

} return null; }

@Override

protected void process(java.util.List<Integer> chunks) {

int latestProgress = chunks.get(chunks.size() - 1);

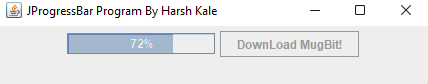
progressBar.setValue(latestProgress); // Update progress bar }

@Override protected void done() { progressBar.setValue(0); // Reset progress bar

startButton.setEnabled(true); // Enable the button after the task is done

} }; worker.execute(); }}

Output:



(Practical 9)

// Programmer: Harsh Moreshwar Kale Created Date: 13/09/2023

// Question 3: Write a program using JProgressBar to show the progress of progressbar when user clicks on JButton.

import javax.swing.\*; import java.awt.event.ActionEvent; import java.awt.event.ActionListener;

public class PracticalNo\_9Q3 {

private static JProgressBar progressBar; private static JButton startButton;

private static JButton pauseButton; private static SwingWorker<Void, Integer> worker;

public static void main(String[] args) {

SwingUtilities.invokeLater(() -> {

JFrame frame = new JFrame("Progress Bar with Start and Pause By Harsh Kale!");

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setSize(300, 100); progressBar = new JProgressBar(0, 100);

progressBar.setStringPainted(true); startButton = new JButton("Download Mugbit!");

startButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) { startProgress(); } });

pauseButton = new JButton("Pause");

pauseButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) { pauseProgress(); } });

JPanel panel = new JPanel();

panel.add(progressBar); panel.add(startButton); panel.add(pauseButton);

frame.add(panel); frame.setVisible(true); }); }

private static void startProgress() {

startButton.setEnabled(false); // Disable the "Start" button while the task is running

pauseButton.setEnabled(true); // Enable the "Pause" button

worker = new SwingWorker<Void, Integer>() {

@Override

protected Void doInBackground() throws Exception {

for (int i = 0; i <= 100; i++) {

if (isCancelled()) { break; }

Thread.sleep(100); publish(i); } return null; }

@Override

protected void process(java.util.List<Integer> chunks) {

int latestProgress = chunks.get(chunks.size() - 1);

progressBar.setValue(latestProgress); // Update progress bar

}

@Override

protected void done() {

progressBar.setValue(0); // Reset progress bar

startButton.setEnabled(true); // Enable the "Start" button

pauseButton.setEnabled(false); // Disable the "Pause" button

}

};

worker.execute();

}

private static void pauseProgress() { if (worker != null) {

worker.cancel(true); // Cancel the task

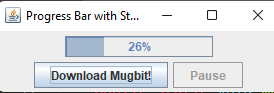
startButton.setEnabled(true); // Enable the "Start" button

pauseButton.setEnabled(false); // Disable the "Pause" button

}

}}

Output:



(Practical 10)

// Programmer: Harsh Moreshwar Kale Created Date: 22/09/2023

// Question 1: Write a Program to demonstrate status of key on Applet Window such as KeyPressed, KeyReleased, KeyUp, KeyDown.

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

public class PracticalNo\_10Q1 extends JFrame implements KeyListener {

    private JLabel keyStatusLabel;

    public PracticalNo\_10Q1() {

        setTitle("Key Status Program in Advance Java by Harsh Kale");

        setSize(300, 100);        setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

        keyStatusLabel = new JLabel("Key Status: ");

        keyStatusLabel.setHorizontalAlignment(JLabel.CENTER);

        getContentPane().add(keyStatusLabel);        addKeyListener(this);

    }

    public void keyTyped(KeyEvent e) {

        keyStatusLabel.setText("Key Typed: " + e.getKeyChar());

    }

    public void keyPressed(KeyEvent e) {

        keyStatusLabel.setText("Key Pressed: " + e.getKeyChar());

    }

    public void keyReleased(KeyEvent e) {

        keyStatusLabel.setText("Key Released: " + e.getKeyChar());

    }

    public static void main(String[] args) {

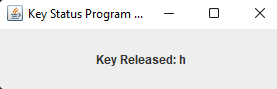
        SwingUtilities.invokeLater(new Runnable() {

            public void run() { System.out.println("Developer Harsh Kale");

                PracticalNo\_10Q1 app = new PracticalNo\_10Q1();

                app.setVisible(true); } }); }}

Output:



(Practical 10)

// Programmer: Harsh Moreshwar Kale Created Date: 22/09/2023

// Question 2: Write a program to generate KeyEvent when a key is pressed and display "KeyPressed" message.

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

public class PracticalNo\_10Q2 {

    public static void main(String[] args) {

        JFrame frame = new JFrame("KeyEvent Simulation Example");

        frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

        frame.setSize(400, 200);

        JPanel panel = new JPanel();        frame.add(panel);

        JButton simulateButton = new JButton("Simulate Key Press");

        panel.add(simulateButton);

        JLabel statusLabel = new JLabel("Status: ");

        panel.add(statusLabel);

        frame.setVisible(true);

        simulateButton.addActionListener(new ActionListener() {

            public void actionPerformed(ActionEvent e) {

                try {

                    System.out.println("Developer Harsh Kale");

                    Robot robot = new Robot();

                    // Press the space key to trigger event!!

                    robot.keyPress(KeyEvent.VK\_A);

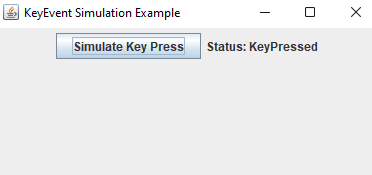
                    robot.keyRelease(KeyEvent.VK\_A);

                    statusLabel.setText("Status: KeyPressed");

                } catch (AWTException ex) {

                    ex.printStackTrace(); }}   });   }}

Output:



(Practical 10)

// Programmer: Harsh Moreshwar Kale Created Date: 22/09/2023

// Question 3: Develop a program which will implement special keys such as function keys and arrow keys.

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

public class PracticalNo\_10Q3 extends JFrame implements KeyListener {

    private JTextArea textArea;

    public PracticalNo\_10Q3() {

        setTitle("Special Keys Example");        setSize(400, 400);

        setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

        textArea = new JTextArea();        textArea.addKeyListener(this);

        getContentPane().add(textArea);    }

    public void keyPressed(KeyEvent e) {

        int keyCode = e.getKeyCode();

        if (keyCode == KeyEvent.VK\_F1) {

            textArea.append("F1 key pressed\n");

        } else if (keyCode == KeyEvent.VK\_F2) {

            textArea.append("F2 key pressed\n");

        } else if (keyCode == KeyEvent.VK\_UP) {

            textArea.append("Up arrow key pressed\n");

        } else if (keyCode == KeyEvent.VK\_DOWN) {

            textArea.append("Down arrow key pressed\n");

        } else if (keyCode == KeyEvent.VK\_LEFT) {

            textArea.append("Left arrow key pressed\n");

        } else if (keyCode == KeyEvent.VK\_RIGHT) {

            textArea.append("Right arrow key pressed\n");

        }    }

    public void keyReleased(KeyEvent e) {

        // Handle keyReleased event (not used in this example)    }

    public static void main(String[] args) {

        SwingUtilities.invokeLater(new Runnable() {

            public void run() {

                System.out.println("Developer Harsh Kale");

                PracticalNo\_10Q3 example = new PracticalNo\_10Q3();

                example.setVisible(true);

            }        });    }

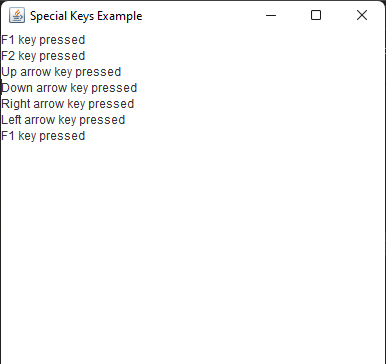
    @Override

    public void keyTyped(KeyEvent e) {

        // TODO Auto-generated method stub

        throw new UnsupportedOperationException("Unimplemented method 'keyTyped' done by harsh kale in future!");    }}

Output:



(Practical 10)

// Programmer: Harsh Moreshwar Kale Created Date: 22/09/2023

// Question 4: Develop a program to accept two numbers and display product of two numbers when user pressed "Multiply" Button.

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

public class PracticalNo\_10Q4 extends JFrame {

    private JTextField num1Field;    private JTextField num2Field;

    private JButton multiplyButton;    private JLabel resultLabel;

    public PracticalNo\_10Q4() {

        setTitle("Multiplication App By Harsh Kale");

        setSize(300, 150);

        setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

        JPanel panel = new JPanel();

        panel.setLayout(new GridLayout(3, 2));

        JLabel num1Label = new JLabel("Enter Number 1:");        num1Field = new JTextField(10);

        JLabel num2Label = new JLabel("Enter Number 2:");        num2Field = new JTextField(10);

        resultLabel = new JLabel("Result: ");

        multiplyButton = new JButton("Multiply");

        multiplyButton.addActionListener(new ActionListener() {

            public void actionPerformed(ActionEvent e) { calculateProduct();}  });

        panel.add(num1Label);        panel.add(num1Field);        panel.add(num2Label);

        panel.add(num2Field);        panel.add(multiplyButton);

        panel.add(resultLabel);

        getContentPane().add(panel);    }

    private void calculateProduct() {

        try {

            double num1 = Double.parseDouble(num1Field.getText());

            double num2 = Double.parseDouble(num2Field.getText());

            double product = num1 \* num2;

            resultLabel.setText("Result: " + product);

        } catch (NumberFormatException ex) {

            resultLabel.setText("Result: Invalid input");

        }

    }

    public static void main(String[] args) {

        SwingUtilities.invokeLater(new Runnable() {

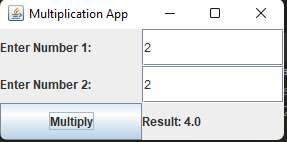
            public void run() {

                System.out.println("Developer Harsh Kale!");

                PracticalNo\_10Q4 app = new PracticalNo\_10Q4();

                app.setVisible(true);            }        });    }}

Output:



(Practical 13)

// Debug the following code and write the output of following code.

import java.awt.\*;

import java.awt.event.\*;

public class PracticalNo\_13Q1 {

Frame f;

PracticalNo\_13Q1() {

f = new Frame("Window Adapter!");

f.addWindowListener(new WindowAdapter() {

public void windowClosing(WindowEvent e) {

f.dispose();

System.exit(0);

}

});

f.setSize(400, 400);

f.setLayout(null);

f.setVisible(true);

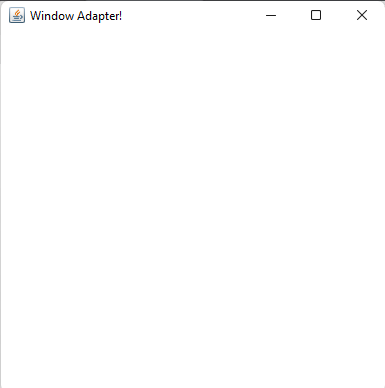
}

public static void main(String[] args) {

new PracticalNo\_13Q1();

}

}

Output:  


(Practical 13)

// Programmer: Harsh Moreshwar Kale

// Created Date: 22/09/2023

// Question 2:

// Write a program to demonstrate the use of WindowAdapter class

import javax.swing.\*;

import java.awt.event.\*;

public class PracticalNo\_13Q2 {

public static void main(String[] args) {

JFrame frame = new JFrame("Window Adapter Program By Harsh Kale!!");

JLabel label = new JLabel("Close the window to exit.");

frame.add(label);

frame.setSize(300, 200);

frame.setDefaultCloseOperation(JFrame.DO\_NOTHING\_ON\_CLOSE);

frame.addWindowListener(new MyWindowAdapter());

frame.setVisible(true);

}

}

class MyWindowAdapter extends WindowAdapter {

@Override

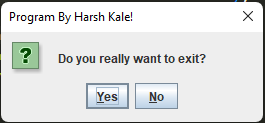
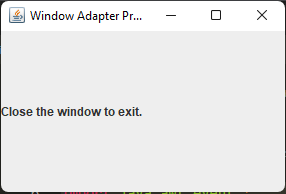
public void windowClosing(WindowEvent e) {

int option = JOptionPane.showConfirmDialog(null, "Do you really want to exit?", "Program By Harsh Kale!", JOptionPane.YES\_NO\_OPTION);

if (option == JOptionPane.YES\_OPTION) {

System.exit(0); } }}

Output:



(Practical 13)

// Programmer: Harsh Moreshwar Kale

// Created Date: 22/09/2023

// Question 3: Write a program to demonstrate the use of anonymous inner class!

import javax.swing.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class PracticalNo\_13Q3 {

public static void main(String[] args) {

JFrame frame = new JFrame("Anonymous Developer Harsh Kale!");

JButton button = new JButton("Click Me");

button.setBounds(100, 50, 150, 30);

button.addActionListener(new ActionListener() {

@Override public void actionPerformed(ActionEvent e) {

JOptionPane.showMessageDialog(null, "Harsh Kale!");

}

});

frame.add(button); frame.setSize(400, 200);

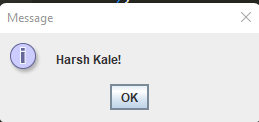
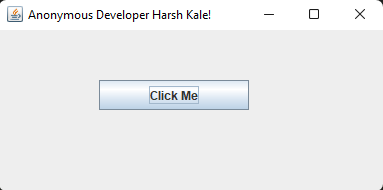
frame.setLayout(null); frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setVisible(true);

}

}

Output:



(Practical 13)

// Programmer: Harsh Moreshwar Kale

// Created Date: 22/09/2023

// Question 4: Write a program using MouseMotionAdapter class to implement only one method mouseDragged().

import javax.swing.\*; import java.awt.\*;

import java.awt.event.MouseAdapter; import java.awt.event.MouseEvent;

public class PracticalNo\_13Q4 {

private JFrame frame;

private int startX, startY, endX, endY;

public PracticalNo\_13Q4() {

frame = new JFrame("Mouse Drag Line Draw Application!");

frame.setSize(400, 400); frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

DrawingPanel drawingPanel = new DrawingPanel();

frame.add(drawingPanel);

drawingPanel.addMouseMotionListener(new MyMouseMotionAdapter());

frame.setVisible(true);

}

class DrawingPanel extends JPanel {

@Override

protected void paintComponent(Graphics g) {

super.paintComponent(g);

g.drawLine(startX, startY, endX, endY);

}

}

class MyMouseMotionAdapter extends MouseAdapter {

@Override

public void mouseDragged(MouseEvent e) {

endX = e.getX();

endY = e.getY();

frame.repaint();

}

}

public static void main(String[] args) {

SwingUtilities.invokeLater(() -> {

System.out.println("Developer Harsh Moreshwar Kale!");

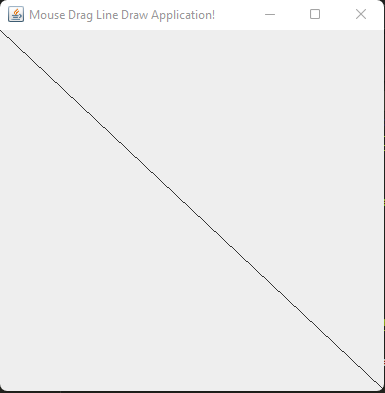
new PracticalNo\_13Q4();

});

}

}

Output:



(Practical No. 14)

// Programmer: Harsh Moreshwar Kale

// Created Date: 25/10/2023

// Question 1:

// Execute the following code and write the output.

import java.net.\*;

public class PracticalNo\_14Q1 {

public static void main(String[] args) {

try{

InetAddress ip = InetAddress.getByName("localhost");

System.out.println("Host Name: " + ip.getHostName());

System.out.println("IP Address: " + ip.getHostAddress());

}catch(Exception e){

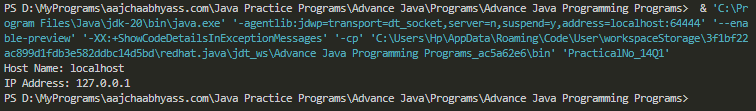
System.out.println(e);

}

}

}

Output:



// Programmer: Harsh Moreshwar Kale

// Created Date: 25/10/2023

// Question 2:

// Develop a program using InetAddress class to retrieve IP address of computer when hostname is entered by the user.

import java.io.IOException; import java.net.\*; import java.util.Scanner;

public class PracticalNo\_14Q2 {

public static void main(String[] args) throws IOException {

System.out.println("Enter any hostname: ");

try (Scanner sc = new Scanner(System.in)) {

String hostname = sc.nextLine();

try {

InetAddress address = InetAddress.getByName(hostname);

if(address.isReachable(3000)){

System.out.println("IP Address: " + address.getHostAddress());

}else{

System.out.println("Host Not Found!");

}

} catch (UnknownHostException e) {

System.out.println("Unknown Host: " + hostname);

}catch(Exception e){

System.out.println("An error occurred: " + e.getMessage());

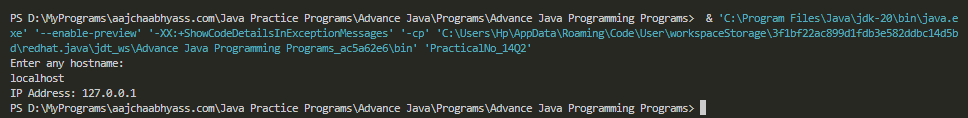
}

}

}

}

Output:



(Practical 15)

// Programmer: Harsh Moreshwar Kale

// Created Date: 25/10/2023

// Question 1:

// Execute the following code and write the output.

import java.net.\*;

public class PracticalNo\_15Q1 {

public static void main(String[] args) throws MalformedURLException {

// URL url = new URL("https://github.com/Harshk133/Advance-Java-Practical-Programs/blob/main/Advance%20Java%20Programming%20Programs/PracticalNo\_14Q2.java");

URL url = new URL("https://www.javatpoint.com/javafx-tutorial");

System.out.println("Protocol: " + url.getProtocol());

System.out.println("Port: " + url.getPort());

System.out.println("Host: " + url.getHost());

System.out.println("File: " + url.getFile());

System.out.println("External form: " + url.toExternalForm());

}

}

Output:



// Programmer: Harsh Moreshwar Kale

// Created Date: 25/10/2023

// Question 2:

// Write a program using URL class to retrieve the host, protocol port and file of URL http://www.msbte.org.in

import java.net.\*;

public class PracticalNo\_15Q2 {

public static void main(String[] args) throws MalformedURLException {

URL url = new URL("http://www.msbte.org.in");

System.out.println("Protocol: " + url.getProtocol());

System.out.println("Port: " + url.getPort());

System.out.println("Host: " + url.getHost());

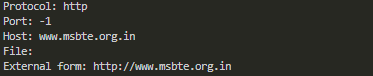
System.out.println("File: " + url.getFile());

System.out.println("External form: " + url.toExternalForm());

}

}

Output:



// Programmer: Harsh Moreshwar Kale

// Created Date: 25/10/2023

// Question 3: Write a program using URL and URLConnection class to retrieve the date, content type, content length information of any entered URL

import java.net.\*;

import java.util.Scanner;

import java.util.Date;

public class PracticalNo\_15Q3 {

public static void main(String[] args) {

System.out.println("Enter any URL: ");

Scanner sc = new Scanner(System.in);

String urlStr = sc.nextLine();

try {

URL url = new URL(urlStr);

URLConnection connection = url.openConnection();

long date = connection.getDate();

Date todayDate = new Date(date);

System.out.println("Date: " + todayDate);

String contentType = connection.getContentType();

System.out.println("Content Type: " + contentType);

int contentLength = connection.getContentLength();

System.out.println("Content length: " + contentLength);

} catch (Exception e) {

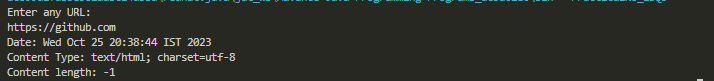
System.out.println("ERROR: " + e.getMessage());

}

}

}

Output:



(Practical No. 16)

// Programmer: Harsh Moreshwar Kale

// Created Date: 25/10/2023

// Question 1:

// Write a program to check credentials of users (Client will send user id and password to server and server will authenticate the client using equals())

// Server-side Program!

import java.io.\*;

import java.net.\*;

public class PracticalNo\_16Q1\_Server {

public static void main(String[] args) {

try {

ServerSocket serverSocket = new ServerSocket(1234);

System.out.println("Server started. Waiting for client connection...");

Socket clientSocket = serverSocket.accept();

System.out.println("Client connected!");

BufferedReader in = new BufferedReader(new InputStreamReader(clientSocket.getInputStream()));

PrintWriter out = new PrintWriter(clientSocket.getOutputStream(), true);

String userId = in.readLine();

String password = in.readLine();

boolean isAuthenticated = authenticate(userId, password);

out.println(isAuthenticated);

in.close();

out.close();

clientSocket.close();

serverSocket.close();

} catch (IOException e) {

e.printStackTrace();

}

}

private static boolean authenticate(String userId, String password) {

return userId.equals("harsh") && password.equals("helloworldprogrammer");

}

}

// Programmer: Harsh Moreshwar Kale

// Created Date: 25/10/2023

// Question 1:

// Write a program to check credentials of users (Client will send user id and password to server and server will authenticate the client using equals())

// Client-side Program!

import java.io.\*;

import java.net.\*;

public class PracticalNo\_16Q1\_Client {

public static void main(String[] args) {

try {

Socket clientSocket = new Socket("localhost", 1234);

System.out.println("Connected to server!!");

BufferedReader in = new BufferedReader(new InputStreamReader(clientSocket.getInputStream()));

PrintWriter out = new PrintWriter(clientSocket.getOutputStream(), true);

BufferedReader userInput = new BufferedReader(new InputStreamReader(System.in));

System.out.print("Enter user ID: ");

String userId = userInput.readLine();

System.out.print("Enter password: ");

String password = userInput.readLine();

out.println(userId);

out.println(password);

boolean isAuthenticated = Boolean.parseBoolean(in.readLine());

if (isAuthenticated) {

System.out.println("Authentication is successfull!!");

} else {

System.out.println("Authentication is failed, Try Again!");

}

in.close();

out.close();

userInput.close();

clientSocket.close();

} catch (IOException e) {

e.printStackTrace();

}

}

}

Output:

Server



Client



// Programmer: Harsh Moreshwar Kale

// Created Date: 25/10/2023

// Question 2:

// Write a program using Socket & ServerSocket to create chat application.

// Server-side Program!

import java.io.\*;

import java.net.\*;

public class PracticalNo\_16Q2ServerChat {

public static void main(String[] args) {

try {

ServerSocket serverSocket = new ServerSocket(1234);

System.out.println("Server started. Waiting for clients...");

Socket clientSocket1 = serverSocket.accept();

System.out.println("Client 1 connected.");

Socket clientSocket2 = serverSocket.accept();

System.out.println("Client 2 connected.");

BufferedReader in1 = new BufferedReader(new InputStreamReader(clientSocket1.getInputStream()));

PrintWriter out1 = new PrintWriter(clientSocket1.getOutputStream(), true);

BufferedReader in2 = new BufferedReader(new InputStreamReader(clientSocket2.getInputStream()));

PrintWriter out2 = new PrintWriter(clientSocket2.getOutputStream(), true);

Thread thread1 = new Thread(new ClientHandler(in1, out2));

Thread thread2 = new Thread(new ClientHandler(in2, out1));

thread1.start();

thread2.start();

serverSocket.close();

} catch (IOException e) {

e.printStackTrace();

}

}

}

class ClientHandler implements Runnable {

private BufferedReader in;

private PrintWriter out;

public ClientHandler(BufferedReader in, PrintWriter out) {

this.in = in;

this.out = out;

}

@Override

public void run() {

try {

String message;

while ((message = in.readLine()) != null) {

System.out.println("Received message: " + message);

out.println(message);

}

} catch (IOException e) {

e.printStackTrace();

}

}

}

// Programmer: Harsh Moreshwar Kale

// Created Date: 25/10/2023

// Question 2:

// Write a program using Socket & ServerSocket to create chat application.

// Client-side Program!

import java.io.\*;

import java.net.\*;

import java.util.Scanner;

public class PracticalNo\_16Q2ClientChat {

public static void main(String[] args) {

try {

try (Socket clientSocket = new Socket("localhost", 1234)) {

System.out.println("Connected to server.");

BufferedReader in = new BufferedReader(new InputStreamReader(clientSocket.getInputStream()));

PrintWriter out = new PrintWriter(clientSocket.getOutputStream(), true);

Thread thread = new Thread(new ServerResponseHandler(in));

thread.start();

try (Scanner scanner = new Scanner(System.in)) {

String message;

while (true) {

message = scanner.nextLine();

out.println(message);

}

}

}

} catch (IOException e) {

e.printStackTrace();

}

}

}

class ServerResponseHandler implements Runnable {

private BufferedReader in;

public ServerResponseHandler(BufferedReader in) {

this.in = in;

}

@Override

public void run() {

try {

String message;

while ((message = in.readLine()) != null) {

System.out.println("Received from server: " + message);

}

} catch (IOException e) {

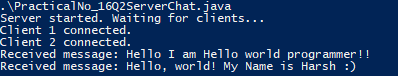
e.printStackTrace();

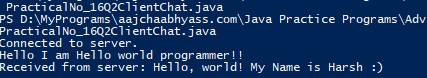
}

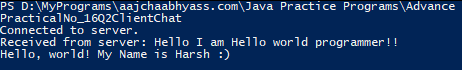
}

}

Output:







// Programmer: Harsh Moreshwar Kale

// Created Date: 25/10/2023

// Question 3:

// Write a program to develop prime number server (Client will send any number to server, Server will send the response the number is prime or not!).

// Server-side Program

import java.io.\*;

import java.net.\*;

public class PracticalNo\_16Q3Server {

public static void main(String[] args) {

try {

try (ServerSocket serverSocket = new ServerSocket(1234)) {

System.out.println("Server started. Waiting for client connection...");

while (true) {

Socket socket = serverSocket.accept();

System.out.println("Client connected: " + socket);

BufferedReader in = new BufferedReader(new InputStreamReader(socket.getInputStream()));

PrintWriter out = new PrintWriter(socket.getOutputStream(), true);

int number = Integer.parseInt(in.readLine());

System.out.println("Number received from client: " + number);

boolean isPrime = checkPrime(number);

if (isPrime) {

out.println(number + " is a prime number");

} else {

out.println(number + " is not a prime number");

}

socket.close();

System.out.println("Client disconnected");

}

} catch (NumberFormatException e) {

e.printStackTrace();

}

} catch (IOException e) {

e.printStackTrace();

}

}

private static boolean checkPrime(int number) {

if (number <= 1) {

return false;

}

for (int i = 2; i <= Math.sqrt(number); i++) {

if (number % i == 0) {

return false;

}

}

return true;

}

}

// Programmer: Harsh Moreshwar Kale

// Created Date: 25/10/2023

// Question 3:

// Write a program to develop prime number server (Client will send any number to server, Server will send the response the number is prime or not!).

// Client-side Program

import java.io.\*;

import java.net.\*;

public class PracticalNo\_16Q3Client {

public static void main(String[] args) {

try {

Socket socket = new Socket("localhost", 1234);

BufferedReader in = new BufferedReader(new InputStreamReader(socket.getInputStream()));

PrintWriter out = new PrintWriter(socket.getOutputStream(), true);

BufferedReader userInput = new BufferedReader(new InputStreamReader(System.in));

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

int number = Integer.parseInt(userInput.readLine());

out.println(number);

String response = in.readLine();

System.out.println("Response from server: " + response);

socket.close();

} catch (IOException e) {

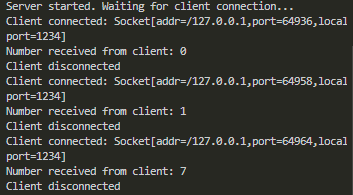
e.printStackTrace();

}

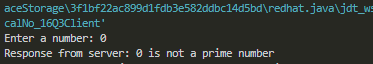
}

}

Output:







(Practical No. 17)

// Programmer: Harsh Moreshwar Kale Created Date: 25/10/2023

// Question 1: Execute the following Program and write the output. One.java

import java.net.\*;

public class PracticalNo\_17Q1One {

public static void main(String[] args) throws Exception {

DatagramSocket ds = new DatagramSocket(3000);

byte[] buf = new byte[1024];

DatagramPacket dp = new DatagramPacket(buf, 1024);

ds.receive(dp);

String str = new String(dp.getData(), 0, dp.getLength());

System.out.println(str);

ds.close(); }}

// Programmer: Harsh Moreshwar Kale Created Date: 25/10/2023

// Question 1: Execute the following Program and write the output. Two.java

import java.net.\*;

public class PracticalNo\_17Q1Two {

public static void main(String[] args) throws Exception {

DatagramSocket ds = new DatagramSocket();

String str = "Harsh shows that Advance Java is Easy!!!";

InetAddress ip = InetAddress.getByName("127.0.0.1");

DatagramPacket dp = new DatagramPacket(str.getBytes(), str.length(), ip, 3000);

ds.send(dp);

ds.close(); }}

Output:



// Programmer: Harsh Moreshwar Kale

// Created Date: 25/10/2023

// Question 2:

// Write a program using DatagramPacket and DatagramSocket to create chat application.

// Server.java

import java.io.\*;

import java.net.\*;

public class PracticalNo\_17Q2Server {

public static void main(String[] args) {

try {

try (DatagramSocket serverSocket = new DatagramSocket(1234)) {

System.out.println("Server Listening & Waiting for client messages...");

while (true) {

byte[] receiveData = new byte[1024];

DatagramPacket receivePacket = new DatagramPacket(receiveData, receiveData.length);

serverSocket.receive(receivePacket);

String clientMessage = new String(receivePacket.getData(), 0, receivePacket.getLength());

System.out.println("Client: " + clientMessage);

String responseMessage = "Server received: " + clientMessage;

byte[] sendData = responseMessage.getBytes();

InetAddress clientAddress = receivePacket.getAddress();

int clientPort = receivePacket.getPort();

DatagramPacket sendPacket = new DatagramPacket(sendData, sendData.length, clientAddress, clientPort);

serverSocket.send(sendPacket);

}

}

} catch (IOException e) {

e.printStackTrace();

}

}

}

import java.io.\*; import java.net.\*;

public class PracticalNo\_17Q2Client {

public static void main(String[] args) {

try {

try (DatagramSocket clientSocket = new DatagramSocket()) {

InetAddress serverAddress = InetAddress.getByName("localhost");

int serverPort = 1234;

BufferedReader userInput = new BufferedReader(new InputStreamReader(System.in));

while (true) {

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

String message = userInput.readLine();

byte[] sendData = message.getBytes();

DatagramPacket sendPacket = new DatagramPacket(sendData, sendData.length, serverAddress, serverPort);

clientSocket.send(sendPacket); byte[] receiveData = new byte[1024];

DatagramPacket receivePacket = new DatagramPacket(receiveData, receiveData.length);

clientSocket.receive(receivePacket);

String serverResponse = new String(receivePacket.getData(), 0, receivePacket.getLength());

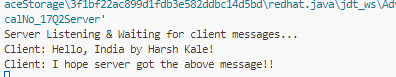
System.out.println("Server: " + serverResponse);

} }

} catch (IOException e) {

e.printStackTrace(); } }}

Output:





// Programmer: Harsh Moreshwar Kale

// Created Date: 25/10/2023

// Question 3:

// Write a program using DatagramPacket and DatagramSocket to copy the contents of one file into other.

// Server.java

import java.io.\*;

import java.net.\*;

public class PracticalNo\_17Q3FileServer {

public static void main(String[] args) {

try {

DatagramSocket serverSocket = new DatagramSocket(1234);

System.out.println("Server started. Waiting for client...");

byte[] receiveData = new byte[1024];

DatagramPacket receivePacket = new DatagramPacket(receiveData, receiveData.length);

serverSocket.receive(receivePacket);

String fileName = new String(receivePacket.getData(), 0, receivePacket.getLength());

File inputFile = new File(fileName);

File outputFile = new File("copy-" + fileName);

FileInputStream fis = new FileInputStream(inputFile);

FileOutputStream fos = new FileOutputStream(outputFile);

byte[] buffer = new byte[1024];

int bytesRead;

System.out.println("Copying file...");

while ((bytesRead = fis.read(buffer)) != -1) {

DatagramPacket sendPacket = new DatagramPacket(buffer, bytesRead, receivePacket.getAddress(), receivePacket.getPort());

serverSocket.send(sendPacket);

fos.write(buffer, 0, bytesRead);

}

fis.close();

fos.close();

serverSocket.close();

System.out.println("File copied successfully.");

} catch (IOException e) {

e.printStackTrace();

}

}

}

// Programmer: Harsh Moreshwar Kale

// Created Date: 25/10/2023

// Question 3:

// Write a program using DatagramPacket and DatagramSocket to copy the contents of one file into other.

// Client.java

import java.io.\*;

import java.net.\*;

public class PracticalNo\_17Q3FileClient {

public static void main(String[] args) {

try {

DatagramSocket clientSocket = new DatagramSocket();

InetAddress serverAddress = InetAddress.getByName("localhost");

int serverPort = 1234;

BufferedReader userInput = new BufferedReader(new InputStreamReader(System.in));

System.out.print("Enter the file name to copy: ");

String fileName = userInput.readLine();

byte[] sendData = fileName.getBytes();

DatagramPacket sendPacket = new DatagramPacket(sendData, sendData.length, serverAddress, serverPort);

clientSocket.send(sendPacket);

FileOutputStream fos = new FileOutputStream("copy-" + fileName);

byte[] buffer = new byte[1024];

System.out.println("Receiving file!!");

while (true) {

DatagramPacket receivePacket = new DatagramPacket(buffer, buffer.length);

clientSocket.receive(receivePacket);

int bytesRead = receivePacket.getLength();

if (bytesRead == 0) {

break;

}

fos.write(buffer, 0, bytesRead);

}

fos.close();

clientSocket.close();

System.out.println("File received successfully.");

} catch (IOException e) {

e.printStackTrace();

}

}

}

Output:







// Programmer: Harsh Moreshwar Kale Created Date: 25/10/2023

// Question 4: Write a program using DatagramPacket and DatagramSocket to transfer the file from one location to another.

import java.io.File; import java.io.FileInputStream;

import java.io.FileOutputStream; import java.net.DatagramPacket;

import java.net.DatagramSocket; import java.net.InetAddress;

public class PracticalNo\_17Q4 {

public static void main(String[] args) {

String sourceFile = "D:\\MyPrograms\\aajchaabhyass.com\\Java Practice Programs\\Advance Java\\Programs\\Advance Java Programming Programs\\Harsh.txt";

String destinationFolder = "C:\\Users\\Hp\\Desktop\\directory\\";

String destinationFile = destinationFolder + "Harsh.txt";

String serverIP = "127.0.0.1"; int serverPort = 12345; int bufferSize = 1024;

try {

FileInputStream fileInputStream = new FileInputStream(sourceFile);

File file = new File(sourceFile);

long fileSize = file.length();

DatagramSocket socket = new DatagramSocket();

InetAddress serverAddress = InetAddress.getByName(serverIP);

byte[] buffer = new byte[bufferSize];

int bytesRead;

int totalBytesSent = 0;

while ((bytesRead = fileInputStream.read(buffer)) != -1) {

DatagramPacket packet = new DatagramPacket(buffer, bytesRead, serverAddress, serverPort);

socket.send(packet);

totalBytesSent += bytesRead;

System.out.println("Sent " + totalBytesSent + " bytes out of " + fileSize + " bytes");

System.out.println("Transfer complete and successful");

}

socket.close();

fileInputStream.close();

FileOutputStream fileOutputStream = new FileOutputStream(destinationFile);

byte[] receiveBuffer = new byte[bufferSize];

DatagramPacket receivePacket = new DatagramPacket(receiveBuffer, bufferSize);

while (true) {

socket.receive(receivePacket);

byte[] data = receivePacket.getData();

fileOutputStream.write(data);

if (receivePacket.getLength() < bufferSize) {

break; } }

fileOutputStream.close(); System.out.println("File transfer completed.");

} catch (Exception e) {

e.printStackTrace(); } }}

Output:

