import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.util.\*; //for Calendar & GregorianCalendar

class MyFrame extends JFrame implements ActionListener, MouseMotionListener, Runnable {

private final Container c;

private final ImageIcon icon;

private final JButton greenBtn,redBtn, yellowBtn,magentaBtn,cyanBtn,lightGrayBtn,grayBtn,pinkBtn,blackBtn,orangeBtn, blueBtn, closeBtn,eraserBtn,restartBtn;

private boolean greenBtnClicked,redBtnClicked,yellowBtnClicked,magentaBtnClicked,cyanBtnClicked, blueBtnClicked;

private boolean lightGrayBtnClicked,grayBtnClicked,blackBtnClicked,orangeBtnClicked,pinkBtnClicked,eraserBtnClicked;

private final Font myFont;

private final Cursor myCursorBefore, myCursorAfter;

private final JLabel label, selectedColor, colorName ;

private JTextArea paintPoint, upperBoundaryLine, lowerBoundaryLine, leftBoundaryLine,rightBoundaryLine,timeArea, dateArea;

MyFrame(String title){

c= this.getContentPane();

c.setLayout(null);

c.setBackground(new Color(230,255,255));

///////instantiate the point where we want to point ///////////

paintPoint = new JTextArea(" ");

///////set title that comes as an argument ////////

this.setTitle(title);

/////set image icon///////////

icon = new ImageIcon("C:\\Users\\zohai\\Desktop\\zohaib.jpg");

this.setIconImage(icon.getImage());

/////////defining My font ///////////

myFont = new Font("Gigi",Font.BOLD,25);

//////////defining cursors///////

myCursorBefore =new Cursor(Cursor.HAND\_CURSOR);

myCursorAfter= new Cursor(Cursor.CROSSHAIR\_CURSOR);

///////////////set label to select color////////////

label = new JLabel("Please select a color:");

label.setBounds(20, 10, 240,30);

label.setFont(myFont);

label.setForeground(Color.DARK\_GRAY);

/////////////selected color label///////////

selectedColor = new JLabel("Selected Button: ");

selectedColor.setBounds(20, 60, 200,30);

selectedColor.setFont(myFont);

selectedColor.setForeground(Color.DARK\_GRAY);

/////////////name of selected color/////////////

colorName = new JLabel("None");

colorName.setBounds(230, 60, 180,30);

colorName.setFont(myFont);

colorName.setForeground(Color.DARK\_GRAY);

///////////////upper boundary line to paint not above this/////////

upperBoundaryLine = new JTextArea("\_");

upperBoundaryLine.setBounds(20, 97, 947,2);

upperBoundaryLine.setBackground(Color.DARK\_GRAY);

upperBoundaryLine.setEditable(false);

///////////////left boundary line to paint not left of this/////////

leftBoundaryLine = new JTextArea("\_");

leftBoundaryLine.setBounds(20, 97, 2,425);

leftBoundaryLine.setBackground(Color.DARK\_GRAY);

leftBoundaryLine.setEditable(false);

///////////////right boundary line to paint not right of this/////////

rightBoundaryLine = new JTextArea("\_");

rightBoundaryLine.setBounds(965, 97, 2,425);

rightBoundaryLine.setBackground(Color.DARK\_GRAY);

rightBoundaryLine.setEditable(false);

//////////////lower boundary line to paint not below this/////////////

lowerBoundaryLine = new JTextArea("\_");

lowerBoundaryLine.setBounds(20, 520, 947,2);

lowerBoundaryLine.setBackground(Color.DARK\_GRAY);

lowerBoundaryLine.setEditable(false);

/////////black button///////////

blackBtn=new JButton("");

blackBtn.setBounds(260,10,30,30);

blackBtn.setFont(myFont);

blackBtn.setCursor(myCursorBefore);

blackBtn.setBackground(Color.BLACK);

blackBtn.setActionCommand("black");

blackBtn.addActionListener(this);

///////////red button////////////

redBtn=new JButton("");

redBtn.setBounds(310,10,30,30);

redBtn.setFont(myFont);

redBtn.setCursor(myCursorBefore);

redBtn.setBackground(Color.RED);

redBtn.setActionCommand("red");

redBtn.addActionListener(this);

/////////yellow button///////////

yellowBtn=new JButton("");

yellowBtn.setBounds(360,10,30,30);

yellowBtn.setFont(myFont);

yellowBtn.setCursor(myCursorBefore);

yellowBtn.setBackground(Color.YELLOW);

yellowBtn.setActionCommand("yellow");

yellowBtn.addActionListener(this);

/////////magenta button///////////

magentaBtn=new JButton("");

magentaBtn.setBounds(410,10,30,30);

magentaBtn.setFont(myFont);

magentaBtn.setCursor(myCursorBefore);

magentaBtn.setBackground(Color.MAGENTA);

magentaBtn.setActionCommand("magenta");

magentaBtn.addActionListener(this);

/////////cyan button///////////

cyanBtn=new JButton("");

cyanBtn.setBounds(460,10,30,30);

cyanBtn.setFont(myFont);

cyanBtn.setCursor(myCursorBefore);

cyanBtn.setBackground(Color.CYAN);

cyanBtn.setActionCommand("cyan");

cyanBtn.addActionListener(this);

/////////lightGray button///////////

lightGrayBtn=new JButton("");

lightGrayBtn.setBounds(510,10,30,30);

lightGrayBtn.setFont(myFont);

lightGrayBtn.setCursor(myCursorBefore);

lightGrayBtn.setBackground(Color.LIGHT\_GRAY);

lightGrayBtn.setActionCommand("light\_Gray");

lightGrayBtn.addActionListener(this);

/////////pink button///////////

pinkBtn=new JButton("");

pinkBtn.setBounds(550,10,30,30);

pinkBtn.setFont(myFont);

pinkBtn.setCursor(myCursorBefore);

pinkBtn.setBackground(Color.PINK);

pinkBtn.setActionCommand("pink");

pinkBtn.addActionListener(this);

/////////gray button///////////

grayBtn=new JButton("");

grayBtn.setBounds(600,10,30,30);

grayBtn.setFont(myFont);

grayBtn.setCursor(myCursorBefore);

grayBtn.setBackground(Color.GRAY);

grayBtn.setActionCommand("gray");

grayBtn.addActionListener(this);

/////////orange button///////////

orangeBtn=new JButton("");

orangeBtn.setBounds(650,10,30,30);

orangeBtn.setFont(myFont);

orangeBtn.setCursor(myCursorBefore);

orangeBtn.setBackground(Color.ORANGE);

orangeBtn.setActionCommand("orange");

orangeBtn.addActionListener(this);

///////////////green button//////////////

greenBtn=new JButton("");

greenBtn.setBounds(700,10,30,30);

greenBtn.setFont(myFont);

greenBtn.setCursor(myCursorBefore);

greenBtn.setBackground(Color.GREEN);

greenBtn.setActionCommand("green");

greenBtn.addActionListener(this);

///////////////blue button//////////////

blueBtn=new JButton("");

blueBtn.setBounds(740,10,30,30);

blueBtn.setFont(myFont);

blueBtn.setCursor(myCursorBefore);

blueBtn.setBackground(Color.BLUE);

blueBtn.setActionCommand("blue");

blueBtn.addActionListener(this);

///////////////time ticking///////////////

timeArea = new JTextArea("");

timeArea.setBounds(780, 10, 203,30);

timeArea.setFont(myFont);

timeArea.setForeground(Color.WHITE);

timeArea.setBackground(new Color(128,0,128));

timeArea.setEditable(false);

///////////////displaying date///////////////

dateArea = new JTextArea("");

dateArea.setBounds(790, 60, 170,30);

dateArea.setFont(myFont);

dateArea.setForeground(Color.WHITE);

dateArea.setBackground(new Color(100,0,200));

dateArea.setEditable(false);

/////////eraser button///////////

eraserBtn=new JButton("Eraser");

eraserBtn.setBounds(420, 60, 100,30);

eraserBtn.setFont(myFont);

eraserBtn.setCursor(myCursorBefore);

eraserBtn.setBackground(Color.DARK\_GRAY);

eraserBtn.setForeground(Color.YELLOW);

eraserBtn.setActionCommand("eraser");

eraserBtn.addActionListener(this);

///////////////close button//////////////

closeBtn=new JButton("Close");

closeBtn.setBounds(535,60,100,30);

closeBtn.setFont(myFont);

closeBtn.setCursor(myCursorBefore);

closeBtn.setBackground(new Color(250,128,128));

closeBtn.setForeground(Color.CYAN);

closeBtn.setActionCommand("close");

closeBtn.addActionListener(this);

/////////restart button///////////

restartBtn=new JButton("Restart");

restartBtn.setBounds(650, 60, 120,30);

restartBtn.setFont(myFont);

restartBtn.setCursor(myCursorBefore);

restartBtn.setBackground(new Color(0,150,250));

restartBtn.setForeground(Color.YELLOW);

restartBtn.setActionCommand("restart");

restartBtn.addActionListener(this);

////////////////adding the components///////////

c.add(label);

c.add(selectedColor);

c.add(colorName);

c.add(upperBoundaryLine);

c.add(leftBoundaryLine);

c.add(rightBoundaryLine);

c.add(lowerBoundaryLine);

c.add(timeArea);

c.add(dateArea);

c.add(greenBtn);

c.add(redBtn);

c.add(yellowBtn);

c.add(cyanBtn);

c.add(blackBtn);

c.add(orangeBtn);

c.add(lightGrayBtn);

c.add(blueBtn);

c.add(grayBtn);

c.add(pinkBtn);

c.add(magentaBtn);

c.add(eraserBtn);

c.add(restartBtn);

c.add(closeBtn);

c.add(paintPoint);

c.addMouseMotionListener(this);

} /////constructor ended//////

//////////////overriding ActionListener's action performed()

// functionality of buttons when clicked///////////

public void actionPerformed(ActionEvent event){

////////to check which button is pressed

String color=event.getActionCommand();

switch (color) {

//////////////////if green button is clicked//////////

case "green":

if (greenBtnClicked) {

makeAllButtonsFalse();

c.setCursor(Cursor.getDefaultCursor());

colorName.setText("None");

colorName.setForeground(Color.DARK\_GRAY);

}

else {

//////////////////make all buttons false ///////////

makeAllButtonsFalse();

/////////////make green btn true/////////

greenBtnClicked=true;

c.setCursor(myCursorAfter);

selectedColor.setText("Selected color: ");

colorName.setText(color.toUpperCase());

colorName.setForeground(Color.green);

}

break;

//////////////////if red button is clicked//////////

case "red":

if (redBtnClicked) {

makeAllButtonsFalse();

c.setCursor(Cursor.getDefaultCursor());

colorName.setText("None");

colorName.setForeground(Color.DARK\_GRAY);

}

else {

//////////////////make all buttons false ///////////

makeAllButtonsFalse();

//////////////make red button true//////////

redBtnClicked = true;

c.setCursor(myCursorAfter);

selectedColor.setText("Selected color: ");

colorName.setText(color.toUpperCase());

colorName.setForeground(Color.red);

}

break;

//////////////////if yellow button is clicked//////////

case "yellow":

if (yellowBtnClicked) {

makeAllButtonsFalse();

c.setCursor(Cursor.getDefaultCursor());

colorName.setText("None");

colorName.setForeground(Color.DARK\_GRAY);

}

else {

//////////////////make all buttons false ///////////

makeAllButtonsFalse();

/////////////make yellow btn true///////////////

yellowBtnClicked = true;

c.setCursor(myCursorAfter);

selectedColor.setText("Selected color: ");

colorName.setText(color.toUpperCase());

colorName.setForeground(Color.yellow);

}

break;

//////////////////if magenta button is clicked//////////

case "magenta":

if (magentaBtnClicked) {

makeAllButtonsFalse();

c.setCursor(Cursor.getDefaultCursor());

colorName.setText("None");

colorName.setForeground(Color.DARK\_GRAY);

}

else {

//////////////////make all buttons false ///////////

makeAllButtonsFalse();

/////////////make magenta btn true/////////

magentaBtnClicked = true;

c.setCursor(myCursorAfter);

selectedColor.setText("Selected color: ");

colorName.setText(color.toUpperCase());

colorName.setForeground(Color.MAGENTA);

}

break;

//////////////////if cyan button is clicked//////////

case "cyan":

if (cyanBtnClicked) {

makeAllButtonsFalse();

c.setCursor(Cursor.getDefaultCursor());

colorName.setText("None");

colorName.setForeground(Color.DARK\_GRAY);

}

else {

//////////////////make all buttons false ///////////

makeAllButtonsFalse();

/////////////make cyan btn true/////////

cyanBtnClicked = true;

c.setCursor(myCursorAfter);

selectedColor.setText("Selected color: ");

colorName.setText(color.toUpperCase());

colorName.setForeground(Color.CYAN);

}

break;

//////////////////if black button is clicked//////////

case "black":

if (blackBtnClicked) {

makeAllButtonsFalse();

c.setCursor(Cursor.getDefaultCursor());

colorName.setText("None");

colorName.setForeground(Color.DARK\_GRAY);

}

else {

//////////////////make all buttons false ///////////

makeAllButtonsFalse();

/////////////make black btn true/////////

blackBtnClicked = true;

c.setCursor(myCursorAfter);

selectedColor.setText("Selected color: ");

colorName.setText(color.toUpperCase());

colorName.setForeground(Color.BLACK);

}

break;

//////////////////if Light-Gray button is clicked//////////

case "light\_Gray":

if (lightGrayBtnClicked) {

makeAllButtonsFalse();

c.setCursor(Cursor.getDefaultCursor());

colorName.setText("None");

colorName.setForeground(Color.DARK\_GRAY);

}

else {

//////////////////make all buttons false ///////////

makeAllButtonsFalse();

/////////////make light gray btn true/////////

lightGrayBtnClicked = true;

c.setCursor(myCursorAfter);

selectedColor.setText("Selected color: ");

colorName.setText(color.toUpperCase());

colorName.setForeground(Color.LIGHT\_GRAY);

}

break;

//////////////////if gray button is clicked//////////

case "gray":

if (grayBtnClicked) {

makeAllButtonsFalse();

c.setCursor(Cursor.getDefaultCursor());

colorName.setText("None");

colorName.setForeground(Color.DARK\_GRAY);

}

else {

//////////////////make all buttons false ///////////

makeAllButtonsFalse();

/////////////make gray btn true/////////

grayBtnClicked = true;

c.setCursor(myCursorAfter);

selectedColor.setText("Selected color: ");

colorName.setText(color.toUpperCase());

colorName.setForeground(Color.GRAY);

}

break;

//////////////////if orange button is clicked//////////

case "orange":

if (orangeBtnClicked) {

makeAllButtonsFalse();

c.setCursor(Cursor.getDefaultCursor());

colorName.setText("None");

colorName.setForeground(Color.DARK\_GRAY);

}

else {

//////////////////make all buttons false ///////////

makeAllButtonsFalse();

/////////////make orange btn true/////////

orangeBtnClicked = true;

c.setCursor(myCursorAfter);

selectedColor.setText("Selected color: ");

colorName.setText(color.toUpperCase());

colorName.setForeground(Color.ORANGE);

}

break;

//////////////////if blue button is clicked//////////

case "blue":

if (blueBtnClicked) {

makeAllButtonsFalse();

c.setCursor(Cursor.getDefaultCursor());

colorName.setText("None");

colorName.setForeground(Color.DARK\_GRAY);

}

else {

//////////////////make all buttons false ///////////

makeAllButtonsFalse();

/////////////make blue btn true/////////

blueBtnClicked = true;

c.setCursor(myCursorAfter);

selectedColor.setText("Selected color: ");

colorName.setText(color.toUpperCase());

colorName.setForeground(Color.BLUE);

}

break;

//////////////////if pink button is clicked//////////

case "pink":

if (pinkBtnClicked) {

makeAllButtonsFalse();

c.setCursor(Cursor.getDefaultCursor());

colorName.setText("None");

colorName.setForeground(Color.DARK\_GRAY);

}

else {

//////////////////make all buttons false ///////////

makeAllButtonsFalse();

/////////////make pink btn true/////////

pinkBtnClicked = true;

c.setCursor(myCursorAfter);

selectedColor.setText("Selected color: ");

colorName.setText(color.toUpperCase());

colorName.setForeground(Color.PINK);

}

break;

//////////////////if pink button is clicked//////////

case "eraser":

if (eraserBtnClicked) {

makeAllButtonsFalse();

c.setCursor(Cursor.getDefaultCursor());

colorName.setText("None");

colorName.setForeground(Color.DARK\_GRAY);

}

else {

//////////////////make all buttons false ///////////

makeAllButtonsFalse();

/////////////make eraser btn true/////////

eraserBtnClicked = true;

c.setCursor(myCursorAfter);

selectedColor.setText("Selected Button: ");

colorName.setText(color.toUpperCase());

colorName.setForeground(Color.DARK\_GRAY);

}

break;

//////////////////////if close btn is clicked////////////////////////

case "close":

System.exit(0);

break;

case "restart":

this.setVisible(false);

int response =JOptionPane.showConfirmDialog(null, "Restart? ","Use Mouse to click on button",JOptionPane.INFORMATION\_MESSAGE);

////////////checking response//////////

switch(response) {

case 0: ////if response is yes///

//////////start new window after half second///////////

try{ Thread.sleep(500); }

catch (InterruptedException ie){ /\*\*/ }

MyFrame frame = new MyFrame("Paint by 19sw42,19sw44,19sw45");

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

frame.setBounds(50, 50, 1000, 600);

frame.setVisible(true);

frame.setResizable(false);

Thread timeThread = new Thread(frame,"Time");

timeThread.start();

break;

case 1: ////////if response is no/////

System.exit(0);

break;

case 2: ////////if response is cancel/////

System.exit(0);

break;

}

break;

} ////switch ended

} ////actionPerformed ended

//////////////overriding MouseMotionListener's methods ()

///////////////////////////mouseMoved() and mouseDragged()////////////////////

public void mouseMoved(MouseEvent event){

}

public void mouseDragged(MouseEvent event){

c.remove(paintPoint); ///so that we can overlap new color over previous color

paintPoint = new JTextArea(" ");

paintPoint.setEditable(false);

int x=event.getX();

int y=event.getY();

try {

///////////if green btn is clicked to paint green color////////////

if(greenBtnClicked) {

if (x >= 23 && x <= 958 && y >= 100 && y <= 513) {

paintPoint.setBounds(x,y, 8, 6);

paintPoint.setBackground(Color.GREEN);

c.add(paintPoint);

}

}

///////////if red btn is clicked to paint red color////////////

if (redBtnClicked){

if (x >= 23 && x <= 958 && y >= 100 && y <= 513) {

paintPoint.setBounds(x,y, 8, 6);

paintPoint.setBackground(Color.RED);

c.add(paintPoint);

}

}

///////////if yellow btn is clicked to paint yellow color////////////

if (yellowBtnClicked){

if (x >= 23 && x <= 958 && y >= 100 && y <= 513) {

paintPoint.setBounds(x,y, 8, 6);

paintPoint.setBackground(Color.YELLOW);

c.add(paintPoint);

}

}

///////////if magenta btn is clicked to paint magenta color////////////

if (magentaBtnClicked){

if (x >= 23 && x <= 958 && y >= 100 && y <= 513) {

paintPoint.setBounds(x,y, 8, 6);

paintPoint.setBackground(Color.MAGENTA);

c.add(paintPoint);

}

}

///////////if cyan btn is clicked to paint cyan color////////////

if (cyanBtnClicked){

if (x >= 23 && x <= 958 && y >= 100 && y <= 513) {

paintPoint.setBounds(x,y, 8, 6);

paintPoint.setBackground(Color.CYAN);

c.add(paintPoint);

}

}

///////////if black btn is clicked to paint black color////////////

if (blackBtnClicked){

if (x >= 23 && x <= 958 && y >= 100 && y <= 513) {

paintPoint.setBounds(x,y, 8, 6);

paintPoint.setBackground(Color.BLACK);

c.add(paintPoint);

}

}

///////////if Light-gray btn is clicked to paint light-gray color////////////

if (lightGrayBtnClicked){

if (x >= 23 && x <= 958 && y >= 100 && y <= 513) {

paintPoint.setBounds(x,y, 8, 6);

paintPoint.setBackground(Color.LIGHT\_GRAY);

c.add(paintPoint);

}

}

///////////if Orange btn is clicked to paint orange color////////////

if (orangeBtnClicked){

if (x >= 23 && x <= 958 && y >= 100 && y <= 513) {

paintPoint.setBounds(x,y, 8, 6);

paintPoint.setBackground(Color.ORANGE);

c.add(paintPoint);

}

}

///////////if Blue btn is clicked to paint blue color////////////

if (blueBtnClicked){

if (x >= 23 && x <= 958 && y >= 100 && y <= 513) {

paintPoint.setBounds(x,y, 8, 6);

paintPoint.setBackground(Color.BLUE);

c.add(paintPoint);

}

}

///////////if Pink btn is clicked to paint pink color////////////

if (pinkBtnClicked){

if (x >= 23 && x <= 958 && y >= 100 && y <= 513) {

paintPoint.setBounds(x,y, 8, 6);

paintPoint.setBackground(Color.PINK);

c.add(paintPoint);

}

}

///////////if gray btn is clicked to paint gray color////////////

if (grayBtnClicked){

if (x >= 23 && x <= 958 && y >= 100 && y <= 513) {

paintPoint.setBounds(x,y, 8, 6);

paintPoint.setBackground(Color.GRAY);

c.add(paintPoint);

}

}

///////////if eraser btn is clicked to erase paint color////////////

if (eraserBtnClicked){

if (x >= 23 && x <= 958 && y >= 100 && y <= 505) {

paintPoint.setBounds(x,y, 10, 15);

paintPoint.setBackground(new Color(230,255,255)); ///background's color

c.add(paintPoint);

}

}

} ///try ended

catch (Exception e){

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

}

} ///ended mouse Dragged

/////////////////////////////own made method for making all buttons false//////////

public void makeAllButtonsFalse(){

greenBtnClicked = false;

redBtnClicked = false;

yellowBtnClicked = false;

magentaBtnClicked = false;

cyanBtnClicked = false;

orangeBtnClicked = false;

blackBtnClicked = false;

blueBtnClicked = false;

grayBtnClicked = false;

lightGrayBtnClicked = false;

pinkBtnClicked = false;

eraserBtnClicked = false;

}

/////////////////overriding run method of Runnable Interface////////

@Override

public void run() {

while(true) {

Calendar time = new GregorianCalendar();

int hour = time.get(Calendar.HOUR);

int minutes = time.get(Calendar.MINUTE);

int seconds = time.get(Calendar.SECOND);

int year = time.get(Calendar.YEAR);

int month= time.get(Calendar.MONTH);

int day= time.get(Calendar.DATE);

int amOrPm= time.get(Calendar.AM\_PM);

if(hour==0) ///as GregorianCalendar has max value 12 for hour

hour=12;

if(amOrPm==1){

timeArea.setText("Time: "+hour + ":" + minutes + ":" + seconds+" PM");

}

else

timeArea.setText("Time: "+hour + ":" + minutes + ":" + seconds+" AM");

dateArea.setText("Date: "+day+"/"+(month+1)+"/"+year);

try { Thread.sleep(1000); }

catch (InterruptedException ie) { /\*\*/ }

} ////////while ended

} ///////////////// run method() ended

}

/////////////////////////////////MyFrame Class ended here///////////////////////

public class OurJavaProject {

public static void main(String[] args) {

MyFrame frame = new MyFrame("Paint by 19sw42,19sw44,19sw45");

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

frame.setBounds(50, 50, 1000, 600);

frame.setVisible(true);

frame.setResizable(false);

Thread timeThread = new Thread(frame,"Time");

timeThread.start();

} //////////////main() ended/////////////

} ///////////Class of main() ended here////////////

**OUTPUT:**

****

**Description:**

This is simple java project made by 19SW42, 19SW44 & 19SW45 that allows user to paint using some colors provided in the top of its interface i.e black, red, yellow, magenta, cyan, light-gray, pink, dark-gray, orange, green and blue. The variables of JFrame, JButton, JLabel, Container, JTextArea and Thread classes’ were used in it, also it has a user defined class MyFrame that extends JFrame and and implements the Interfaces ActionListener, MouseMotionListener and Runnable by providing implementations for the methods of these interfaces in its body. The program’s interface has also date and time areas that display the current date and time.

To use the above classes in this program

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

, these packages were imported.

As this program works like paint software hence its title is “Paint by 19sw42, 19sw44, 19sw45” and this was all about our simple project.

**NOTE: DEMO VIDEO IS ATTACHED IN THE SAME .ZIP FOLDER.**