Experiment 6 Music Player Implementation

【Purpose】

1. Learn Java Graphical User Interface Design

2. Learn java event handling

【Experimental principle】

1. Fundamentals of Java programming

2. Fundamentals of object-oriented programming

【Experimental content】

Experimental content : Design a complete player software according to the example program of the player .

Experimental requirements:

1 Design a complete player interface and add button responses;

2 Realize that the lyrics are reminded sentence by sentence according to the content of the song ;

3 Realize that the progress bar moves according to the playing time of the song ;

4 Import a gif animation picture ;

5 Add a column of song time on the right side of the song list ;

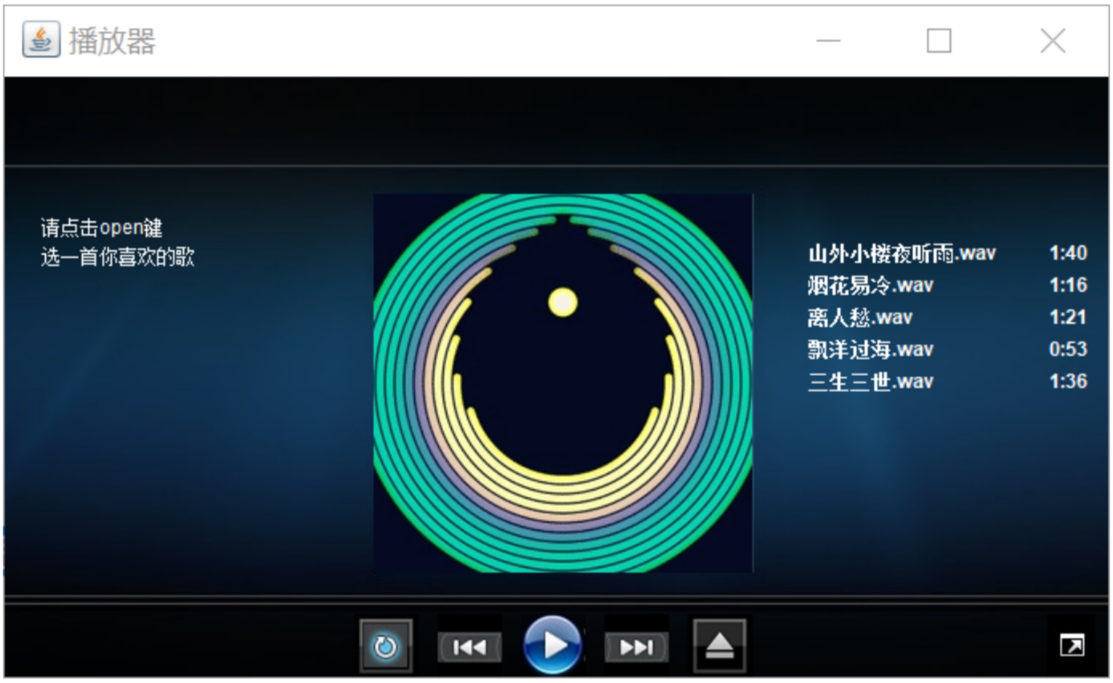
6 The progress bar attachment adds a display of the current playing time ;

7 Double-click the list to play the song;

8 Add the functions you think you need. . .

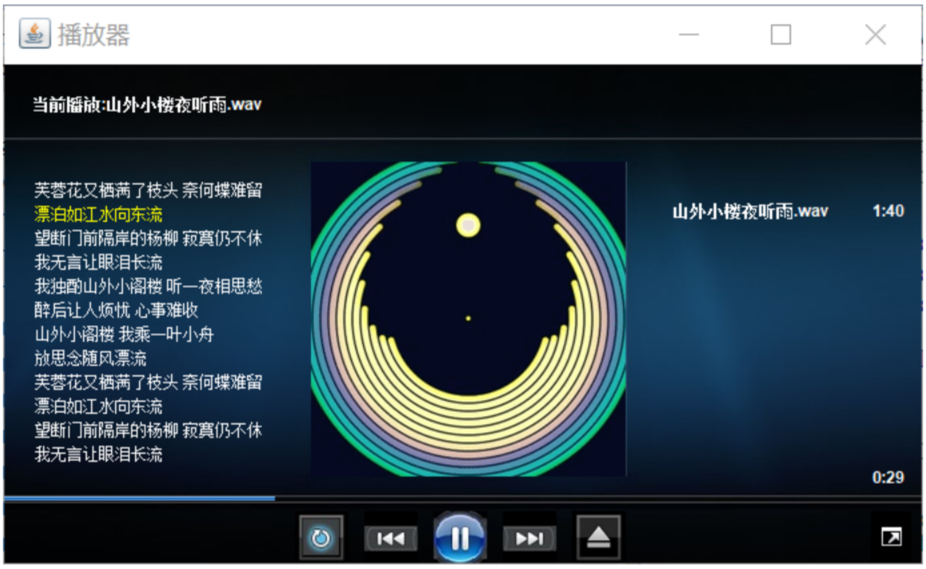
9 Finally, pack the player program into a jar file .

My code comes from PlayMusic1.java in src in the compressed package



请点击open键
选一首你喜欢的歌

山外小楼夜听雨.wav	1:40
烟花易冷.wav	1:16
离人愁.wav	1:21
飘洋过海.wav	0:53
三生三世.wav	1:36



当前播放:山外小楼夜听雨.wav

芙蓉花又栖满了枝头 奈何蝶难留
漂泊如江水向东流
望断门前隔岸的杨柳 寂寞仍不休
我无言让眼泪长流
我独酌山外小阁楼 听一夜相思愁
醉后让人烦忧 心事难收
山外小阁楼 我乘一叶小舟
放思念随风漂流
芙蓉花又栖满了枝头 奈何蝶难留
漂泊如江水向东流
望断门前隔岸的杨柳 寂寞仍不休
我无言让眼泪长流

山外小楼夜听雨.wav 1:40

0:29

```
import java.applet.Applet;
import java.applet.AudioClip;
import java.net.MalformedURLException;
import java.net.URL;
import java.io.*;
import javax.swing.text.*;
import javax.swing.text.StyleContext.NamedStyle;
import javax.swing.*;
import static javax.swing.JFrame.*; //Introduce static constants of JFrame
import java.awt.event.*;
import java.awt.*;
import java.net.*;
import java.util.*;
import java.util.Timer;
```

```
class audioplay{//play music class
    AudioClip adc;// sound audio clip object
    URL url;
    boolean adcFlag=false;
    boolean playFlag=false;
    void SetPlayAudioPath(String path){
          try{
               url = new URL(path);
              // System.out.println(adc.toString());
               if(adcFlag==true){adc.stop();playFlag=false;}
               adc = Applet.newAudioClip(url);
               adcFlag=true;
             }
          catch (MalformedURLException e1) {
                  e1.printStackTrace();
             }
       }
       void play(){      //Start playing
               adc.play();
               playFlag=true;
       }
       void stop(){      //Stop play
               adc.stop();
               playFlag=false;
       }
}

/*
class music
{
    //LinkedList<String> Lyricslist;

    LinkedList<String> Lyricslist=new LinkedList<String>();//Song singer and lyrics
information
    LinkedList<String> Lyrics=new LinkedList<String>();//lyric information
    LinkedList<Integer> Lyricstime=new LinkedList<Integer>();//Lyrics time
information
    String name;//song name
    String songer;//singer
    String time;//Total duration
    String playFileDirectory;//file path
    //String[] Lyrics=new String[100];//lyric information
//int[] Lyricstime=new int[((Lyricslist.size()-1)/3)*2];//Lyrics time information
    music(String na,String ti)
```

```
{
    this.name=na;
    this.time=ti;
}
void loadlyrics(String path,String name) {
    //The parameters are the path of the song and the name of the song
    //Import singer and lyrics
    int n=0;
    String Lyricspath=path+name+".txt";//Find the lyrics path
    File filename = new File(Lyricspath);
    InputStreamReader reader = null;
    try {
        reader = new InputStreamReader(new FileInputStream(filename));
    } catch (FileNotFoundException e1) {}
    BufferedReader br = new BufferedReader(reader);
    String line="";
    String[] split= {"","",""};

    try {
        while ((line = br. readLine()) != null ) {
// read data one row at a time
            if(n==0) {
                Lyricslist.add(line);//Singer name
    //System.out.println(Lyricslist.get(n));
    n++;
            }
            else {
                //Lyricslist.add(line);
    //System.out.println(Lyricslist.get(n));
    split=line. split(" ");
    Lyricslist.add(split[0]);//Start time of each line of lyrics
    Lyricslist.add(split[1]);//The end time of each line of lyrics
    Lyricslist.add(split[2]);//Each line of lyrics
    n++;
    //System.out.println(Lyricslist.size());
            }
        }
     } catch (IOException e1) {}

     int x=0;
     for(x=0;x<(Lyricslist.size()-1)/3;x++) {
     //System.out.println(Lyricslist.get(x*3+1));
     //System.out.println(Lyricslist.size());
     Lyricstime.add(Integer.parseInt(Lyricslist.get(x*3+1)));
     Lyricstime.add(Integer.parseInt(Lyricslist.get(x*3+2)));
```

```
         //System.out.println(Lyricstime[x*2]+"+"+Lyricstime[x*2+1]);
         Lyrics.add(Lyricslist.get(x*3+3)+"\n");
         //System.out.println(Lyrics.get(x));
}//Each lyric and its start and end time
    }
    int timetransform(String s){//Convert song time into seconds
        String[]t={"",""};
        t=s.split(":");
        return Integer. parseInt(t[0])+Integer. parseInt(t[1]);
    }
}
*/

class MyExtendsJFrame extends JFrame implements ActionListener,MouseListener{//window
class
    JLabel background;//Background control

    JButton buttonPlay;//play button
    JButton buttonPrev;
    JButton buttonNext;
    JButton buttonGif;
    JButton buttonUnloop;
    JButton buttonOpenFile;
    JButton buttonList;

    audioplay audioPlay;


    JTextPane textLyrics;//lyric control
    JLabel playTime;//play progress bar control
    JList listPlayFile;//playlist control
    JList listTimeFile;
    JList timenow;
    JList songnow;
    Timer nTimer;//timer object

    JButton closeBtn = null;
    JButton maxBtn = null;
    JButton minBtn = null;

    public MyExtendsJFrame(){

    audioPlay = new audioplay();

        setTitle("player");//software name
        setBounds(160,100,710,430);   //Set window size
        setLayout(null);//empty layout
        init(); //The operation of adding controls is encapsulated into a function
        setVisible(true);//Execute after adding components
        setDefaultCloseOperation(EXIT_ON_CLOSE);
    }




    void init(){//Added control
        Icon img=new ImageIcon(”.//background.jpg”); //Create icon object

        background = new JLabel(img);//Set the background image
        background.setBounds(0,0,700,400);//Set background control size
    getLayeredPane().add(background, new Integer(Integer.MIN_VALUE));//The background picture control is placed at the bottom
        ((JPanel)getContentPane()).setOpaque(false); //The control is transparent


        buttonPlay=new JButton();//Add play button
    buttonPlay.setBounds(322,335,40,40); //Set the size of the play button
    //Icon icon=new ImageIcon(”.//play.jpg”);//Create a player icon object
    //Icon icon2=new ImageIcon(”.//stop.jpg”);
    buttonPlay.setIcon(icon);     //Set the play icon
    buttonPlay.setBorderPainted(false); //button border transparent

    Icon pressedIcon=new ImageIcon(”.//play2.jpg”); //Create an icon object when clicked
    buttonPlay.setPressedIcon(pressedIcon); //Set the icon when clicked
    //buttonPlay.setBorderPainted(false); //Button border becomes transparent when clicked
    //Icon overIcon=new ImageIcon(”.//stop.jpg”);
    //buttonPlay.setRolloverIcon(overIcon);
    buttonPlay.addActionListener(this);//Add click event association
    add(buttonPlay);//add the play button to the window


        buttonPrev=new JButton();//Add play button
    buttonPrev.setBounds(270,335,40,40); //Set the size of the play button
    Icon PrevIcon=new ImageIcon(”.//prev.jpg”);//Create a playback icon object
    buttonPrev.setIcon(PrevIcon);     //Set the playback icon
    buttonPrev.setBorderPainted(false); //button border transparent
    Icon pressedPrevIcon=new ImageIcon(”.//prev3.jpg”);
```

```
buttonPrev.setPressedIcon(pressedPrevIcon);
buttonPrev.setBorderPainted(false);
buttonPrev. addActionListener(this);
add(buttonPrev);//add the play button to the window

    buttonNext=new JButton();//Add play button
buttonNext.setBounds(374,335,40,40); //Set the size of the play button
Icon NextIcon=new ImageIcon(″.//next.jpg″);//Create a player icon object
buttonNext.setIcon(NextIcon);     //Set the play icon
buttonNext.setBorderPainted(false); //button border transparent
//Icon pressedNextIcon=new ImageIcon(″.//next2.jpg″);
//buttonPlay.setPressedIcon(pressedNextIcon);
//buttonPlay.setBorderPainted(false);
buttonNext. addActionListener(this);
add(buttonNext);//add the play button to the window

    buttonGif=new JButton();//Add play button
buttonGif.setBounds(230,73,236,236); //Set the size of the play button
Icon gif3=new ImageIcon(″.//gif9.png″);
buttonGif.setIcon(gif3); //Set the play icon
buttonGif.setBorderPainted(false); //button border transparent
buttonGif. addActionListener(this);
add(buttonGif);//add the play button to the window

    buttonUnloop=new JButton();//Add play button
buttonUnloop.setBounds(218,335,40,40); //Set the size of the play button
Icon unloop=new ImageIcon(″.//unloop.jpg″);//Create a playback icon object
buttonUnloop.setIcon(unloop);     //Set the playback icon
buttonUnloop.setBorderPainted(false); //button border transparent
buttonUnloop. addActionListener(this);
add(buttonUnloop);//add the play button to the window

buttonOpenFile=new JButton();//Add play button
buttonOpenFile.setBounds(426,335,40,40); //Set the size of the play button
Icon open=new ImageIcon(″.//open.jpg″);//Create a player icon object
buttonOpenFile.setIcon(open);     //Set the playback icon
buttonOpenFile.setBorderPainted(false); //button border transparent
buttonOpenFile.addActionListener(this);//Add click event association
add(buttonOpenFile);//add the play button to the window

    buttonList=new JButton();//Add play button
buttonList.setBounds(650,335,30,40); //Set the size of the play button
Icon list=new ImageIcon(″.//list.jpg″);//Create a player icon object
buttonList.setIcon(list);     //Set the play icon
buttonList.setBorderPainted(false); // button border transparent
```

```
    add(buttonList);//add play button to the window


    listPlayFile=new JList();     //Create a playlist
    listPlayFile.setBounds(500,100,150,150); //Set playlist size
    listPlayFile.setOpaque(false);//Set playlist transparency
    listPlayFile.setBackground(new Color(0, 0, 0, 0));//Set the background color of the playlist
    listPlayFile.setForeground(Color.white);//Set the font color of the playlist
    add(listPlayFile); //add the playlist to the window
    listPlayFile.addMouseListener(this);//Add the double-click event association of the playlist
        //vt1.add("The small building outside the mountain listens to the rain at night.wav");
        //vt1.add("Fireworks are easy to cool.wav");
        //vt1.add("Worry about leaving people.wav");
        //vt1.add("Crossing the ocean.wav");
        //vt1.add("Three Lives Three Worlds.wav");
        //listPlayFile.setListData(vt1);//Add the Vector object to the playlist control

    listTimeFile = new JList();
    listTimeFile.setBounds(650,100,40,150);
    listTimeFile.setOpaque(false);//Set playlist transparency
    listTimeFile.setBackground(new Color(0, 0, 0, 0));//Set the background color of the playlist
    listTimeFile.setForeground(Color.white);//Set the font color of the playlist
    add(listTimeFile); //add the playlist to the window
    listTimeFile.addMouseListener(this);//Add the double-click event association of the playlist




    timenow=new JList();
    timenow.setBounds(650,300,30,20);
    timenow.setOpaque(false);//Set playlist transparency
    timenow.setBackground(new Color(0, 0, 0, 0));
    timenow.setForeground(Color.white);
    add(timenow);

    songnow=new JList();
    songnow.setBounds(20,20,200,50);
    songnow. setOpaque(false);
    songnow.setBackground(new Color(0, 0, 0, 0));
    songnow.setForeground(Color.white);
    add(songnow);

    Icon img2=new ImageIcon(".//time.jpg"); //Create icon object
    playTime = new JLabel(img2);          //Create a playback progress bar object
    playTime. setBounds(0,324,0,3);       //Set the size of the playback progress bar
object
    add(playTime); //Add playback progress bar to the window

    /*
        //Set the maximize button
        maxBtn = new JButton("");
        Icon icon6=new ImageIcon(".//max.png");
        maxBtn.setIcon(icon6);
        maxBtn. setBounds(645, 8, 24, 24);
        maxBtn.setBorderPainted(false);
    maxBtn.setHorizontalAlignment(JButton.CENTER);
    maxBtn. addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {

    setExtendedState(JFrame.MAXIMIZED_BOTH);//Maximize the form

    }
    });
    this. add(maxBtn);
    //Set the minimize button
    minBtn = new JButton("");
    Icon icon7=new ImageIcon(".//min.png");
        minBtn.setIcon(icon7);
        minBtn.setBounds(615, 8, 24, 24);
        minBtn.setBorderPainted(false);
    minBtn.setHorizontalAlignment(JButton.CENTER);
    minBtn. addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {

    setExtendedState(JFrame.ICONIFIED);//Minimize the form

    }
    });
    this. add(minBtn);
    */
    textLyrics=new JTextPane(); //Create lyrics control
    textLyrics.setBounds(20,80,200,220);//Set the size of the lyrics control
    textLyrics.setForeground(Color.white);//The font color of the lyrics control
    textLyrics.setOpaque(false);//The lyrics control is transparent
           add(textLyrics);    //添加歌词控件至窗口中




    }


    public static void read2(String filePath){
        System.out.println(″------second method-------″);
        File file = new File(filePath);
        if(file.exists()){
            try {
                FileReader fileReader = new FileReader(file);
                BufferedReader br = new BufferedReader(fileReader);
                String lineContent = null;
                while((lineContent = br.readLine())!=null){
                    System.out.println(lineContent);
                }
                br.close();
fileReader. close();
} catch (FileNotFoundException e) {
System.out.println(″no this file″);
e.printStackTrace();
} catch (IOException e) {
System.out.println(″io exception″);
e.printStackTrace();
}
}
}


Icon icon=new ImageIcon(″.//play.jpg″);//Create a player icon object
Icon icon2=new ImageIcon(″.//stop.jpg″);
Icon gif=new ImageIcon(″.//playgif7.gif″);//Create a player icon object
Icon gif2=new ImageIcon(″.//playgif6.gif″);

    String[] text1=new String[]{
            ″The hibiscus flowers are full of branches again, but the butterflies are
```

```
hard to stay\n",
            "Wandering like a river flowing eastward\n",
            "Looking at the willows across the bank in front of the broken gate is still lonely\n",
            "I am speechless and let the tears flow forever\n",
            "I drink alone in the small attic outside the mountain and listen to the lovesickness all night\n",
            "Drunk makes people worry and worry hard\n",
            "In the small attic outside the mountain, I will take a small boat\n",
            "Let’s miss and drift with the wind\n",
            "The hibiscus flowers are full of branches again, but the butterflies are hard to stay\n",
            "Wandering like a river flowing eastward\n",
            "Looking at the willows across the bank in front of the broken gate is still lonely\n",
            "I am speechless and let the tears flow forever\n",
    };

    public void LyricsMove(String address){

        File file=new File(address);
int iMusicTime=(int)file.length()/1024/173;
    }
    int[] ti=new int[]{0,23,30,37,43,52,58,65,71,79,85,93};

    public void timerFun(final int t,final int f){//timer function
        if(nTimer!=null){nTimer.cancel();}//If there is already a timer, it will be closed
nTimer = new Timer();//Create a timer
    if(audioPlay.playFlag !=false)
nTimer.schedule(new TimerTask(){ //Anonymous class
        int nPlayTime=0;
        String timeshow;

public void run() { //timer function body
            playTime.setBounds(0, 324, nPlayTime+=t, 3);
            Vector vt=new Vector (); //Create a Vector object and add multiple lines through the add method
            if ((nPlayTime/t)<f ){

            if ((nPlayTime/t)<10)
            vt.add("0:0"+(nPlayTime/t));
    else if((nPlayTime/t)<60)
            vt.add("0:"+(nPlayTime/t));
    else if((nPlayTime/t)>=60)
```

```
vt.add("1:"+((nPlayTime/t)-60));
timenow.setListData(vt);
      }
      else
      vt.add("1:"+(f-60));
      int i=0,a=0;
     StyledDocument doc = textLyrics.getStyledDocument();
     Style style = textLyrics.addStyle("I'm a Style", null);
     //System.out.println(nPlayTime/t);
     if((nPlayTime/t)<f)
     {

      for(int m=0;m<12;m++)
      {

              if(((nPlayTime/t)>ti[m])&&(nPlayTime/t<ti[m+1]))
              {
              i=m;
                          //System.out.println(i);
              break;

              }

      }
            if(i==a+1){
                  textLyrics.setText(null);
                  a=i;
            }
            for(int j=0;j<12;j++)
            {

              if(j==i)
              {
                          StyleConstants.setForeground(style, Color.yellow);
                          try
                          {
                           doc.insertString(doc.getLength(), text1[i],style);
                          }
                          catch (BadLocationException e){}
              }
                   else if(j!=i)
                   {
                         StyleConstants.setForeground(style, Color.white);
                         try
                         {
```

```
                doc.insertString(doc.getLength(), text1[j],style);
    }
    catch (BadLocationException e){}
    }
    getContentPane().add(textLyrics);
    setVisible(true);
        }

}

}
},0,1000);
}


    //@SuppressWarnings("unchecked")//Ignore warnings

    String song1="Listen to the rain at night in the small building outside the
mountain.wav";
    //String song2="Fireworks are easy to be cold.wav";
    //String song3="Worry about leaving people.wav";
    //String song4="Across the Ocean.wav";
    //String song5="Three Lives Three Worlds.wav";

    String playFileName = song1;
    String playFile;
    String playFileDirectory;

Vector vt1=new Vector(); //Create a Vector object and add multiple lines through the
add method


//private JTextPane textPanel;
/*
private StyleContext styleContext;
private DefaultStyledDocument doc;

public void initCompoment() {
    styleContext = new StyleContext();
    doc = new DefaultStyledDocument(styleContext);

    textLyrics=new JTextPane(doc); //Create lyrics control
    textLyrics.setBounds(20,80,200,220);//Set the size of the lyrics control
    textLyrics.setForeground(Color.white);//The font color of the lyrics control
    textLyrics.setOpaque(false);//The lyrics control is transparent
    add(textLyrics); //add lyrics control to the window
    //textLyrics.setText("The hibiscus flower is full of branches again\n"+"But it's
hard for the butterfly to stay\n"+
//          "Wandering like a river flowing eastward\n"+"Looking at the willows across
the bank in front of the broken gate\n");//Add text to the lyrics control
    textLyrics. setText(
    "Please click the open button\n"+"Choose a song you like\n"
    );

    final Style greenStyle = styleContext.addStyle("ConstantWidth", null);
    StyleConstants.setFontFamily(greenStyle, "monospaced");
    StyleConstants.setForeground(greenStyle, Color.green); // set yellow text

    final Style yellowStyle = styleContext.addStyle("ConstantWidth", null);
    StyleConstants.setFontFamily(yellowStyle, "monospaced");
    StyleConstants.setForeground(yellowStyle, Color.yellow);   // set blue text

    final Style blueStyle = styleContext.addStyle("ConstantWidth", null);
    StyleConstants.setFontFamily(blueStyle, "monospaced");
    StyleConstants.setForeground(blueStyle, Color.blue);

    try {
    SwingUtilities. invokeAndWait(new Runnable() {
    public void run() {
    try {
    // add text to the document
    doc.insertString(0, text1, null);
    // set the first segment to green
    doc. setParagraphAttributes(0, 1, greenStyle, false);
    // set the second segment to yellow
    doc.setParagraphAttributes(12, 1, yellowStyle, false);
    // Set the third and fourth paragraphs to blue
    doc.setParagraphAttributes(25, 13, blueStyle, false); }                catch
(BadLocationException e) {
    }
}
*/


    public void timeModify(){
        if (playFileName. contentEquals(song1))
        {
    timerFun(7,100);//Open the timer
    //textLyrics.setText(text1);
        }
        /*
        else if(playFileName. contentEquals(song2)){
            timerFun(9,76);
    textLyrics. setText(text2);
        }
        else if(playFileName. contentEquals(song3)){
             timerFun(9,81);
             textLyrics.setText(text3);
         }
         else if(playFileName.contentEquals(song4)){
             timerFun(13,53);
             textLyrics.setText(text4);
         }
         else if(playFileName.contentEquals(song5)){
             timerFun(7,96);
             textLyrics.setText(text5);
         }
         else
             timerFun(200,1000);
             */

         Vector vv=new Vector();
         vv.add("currently playing:"+playFileName);
         songnow.setListData(vv);
    }

    int i=0;

    public void actionPerformed(ActionEvent e){

        if(e.getSource()==buttonOpenFile){//If it is an open file button click event
            FileDialog openFile=new FileDialog(this,"Open File");//Create a dialog
box to open a file
            openFile.setVisible(true);//The dialog box is visible
            playFileName=openFile.getFile();//Get the open file name
            playFileDirectory=openFile.getDirectory();//Get the opened file path
            playFile=playFileDirectory+playFileName;//Complete path + file name
            audioPlay.SetPlayAudioPath("file:"+playFile);//Set the playback file
            audioPlay.play();//Start playing
        buttonPlay.setIcon(icon2);

            vt1.add(playFileName);
            //vt1.add("Fireworks are easy to cool.wav");
            //vt1.add("Worry about leaving people.wav");
            //vt1.add("Crossing the ocean.wav");
            //vt1.add("Three Lives Three Worlds.wav");
            listPlayFile.setListData(vt1);//Add the Vector object to the playlist
control
        //System.out.println(playFile);

        Vector vtt=new Vector ();    //Create a Vector object and add multiple lines
through the add method
        vtt.add("1:40");
        //vtt.add("1:16");
        //vtt.add("1:21");
        //vtt.add("0:53");
        //vtt.add("1:36");
        listTimeFile.setListData(vtt);//Add the Vector object to the playlist control

        buttonGif.setIcon(gif);  //Set the play icon

            timeModify();
            //read2("file:"+playFileDirectory +"text1.txt");

        }



        if(e.getSource()==buttonPlay){//If it is a play button click event

            if(audioPlay.playFlag==false){
            audioPlay.SetPlayAudioPath("file:"+playFile);//Set the playback file
            audioPlay.play();
        buttonPlay.setIcon(icon2);
            timeModify();
            }
            else if(audioPlay.playFlag==true)
            {
                audioPlay. stop();
                buttonPlay.setIcon(icon);
                timeModify();
            }

             System.out.println(audioPlay.playFlag);
        }

        if(e.getSource()==buttonPrev){//If it is a play button click event
            /*
            if(playFileName.contentEquals(song1))
                playFileName=song5;
            else if(playFileName.contentEquals(song2))
```

```
                playFileName=song1;
            else if(playFileName.contentEquals(song3))
                playFileName=song2;
            else if(playFileName.contentEquals(song4))
                playFileName=song3;
            else if(playFileName.contentEquals(song5))
                playFileName=song4;
                */

             //audioPlay.SetPlayAudioPath("file:.\\"+playFileName);//设置播放文件
             playFile=playFileDirectory+playFileName;
             audioPlay.SetPlayAudioPath("file:"+playFile);//set play file
             audioPlay.play();
             buttonPlay.setIcon(icon2);

             timeModify();


        }
        if(e.getSource()==buttonNext){
            /*
            if(playFileName.contentEquals(song1))
                playFileName=song2;
            else if(playFileName.contentEquals(song2))
                playFileName=song3;
            else if(playFileName.contentEquals(song3))
                playFileName=song4;
            else if(playFileName.contentEquals(song4))
                playFileName=song5;
            else if(playFileName.contentEquals(song5))
                playFileName=song1;
                */

            //audioPlay.SetPlayAudioPath("file:.\\"+playFileName);//Set         the
playback file
            playFile=playFileDirectory+playFileName;
            audioPlay.SetPlayAudioPath("file:"+playFile);//Set the playback file
            audioPlay.play();
        buttonPlay.setIcon(icon2);

            timeModify();
        }
        if(e.getSource()==buttonUnloop){
            audioPlay.SetPlayAudioPath("file:"+playFile);//Set the playback file
            audioPlay.play();
        buttonPlay.setIcon(icon2);
            timeModify();
        }

    }

    public void mousePressed(MouseEvent e){}//press the mouse
        public void mouseReleased(MouseEvent e){}//Release the mouse
        public void mouseEntered(MouseEvent e){}//mouse enters
        public void mouseExited(MouseEvent e){}//mouse leaves
        public void mouseClicked(MouseEvent e){//Click the mouse
            if (e.getClickCount() == 2) {//If the mouse clicks twice in a row
                if(e.getSource()==listPlayFile){//If the event source is a playlist，that is，double-click in the playlist control，execute.
                    int index=listPlayFile.getSelectedIndex();
                    //System.out.println(index);
                    String str=(String) vt1. get(index);
                    System.out.println(str);

                    playFileName = str;
                    System.out.println(playFileName);

                    timeModify();
                    playFile=playFileDirectory+playFileName;
                    audioPlay.SetPlayAudioPath("file:"+playFile);//Set the playback file
                    audioPlay.play();
                buttonPlay.setIcon(icon2);
                buttonGif.setIcon(gif2);
                    //Add the code in the double-click playlist，such as getting the song name and playing it.
                }
            }
    }

    /*
    class JButtonHandler implements ActionListener {//Listener class 2
        public void actionPerformed(ActionEvent e){
            System.out.println("Action occurred");
        }
    }
public void actionPerformed(ActionEvent e){    //action event function
    if(e.getSource()==buttonOpenFile){
    FileDialog openFile=new FileDialog(this,"Open File");//Create a dialog box to open a file
    openFile.setVisible(true);//The dialog box is visible
    String playFileName=openFile.getFile();//Get the open file name
    String playFileDirectory=openFile.getDirectory();//Get the opened file path
    String playFile=playFileDirectory+playFileName;//Complete path + file name
    audioPlay.SetPlayAudioPath("file:"+playFile);//Set the playback file
    audioPlay.play();//Start playing
    }
}
public void actionPerformed(ActionEvent e){   //action event function
    if(e.getSource()==buttonPlay){
    audioPlay.SetPlayAudioPath("file:"+".\\The small building outside the mountain listens to the rain at night.wav");
    audioPlay.play();
    timerFun();
    }
}
*/

}

public class MusicPlay1{
//@SuppressWarnings("unchecked")//Ignore warnings
public static void main(String[] args){
    audioplay audioPlay=new audioplay(); //Create a playback object
    audioPlay.SetPlayAudioPath("file:"+".\\The small building outside the mountain listens to the rain at night.wav");//Set the playback file
        // audioPlay.play();//Start playing
MyExtendsJFrame frame=new MyExtendsJFrame();//Create a chat program window

//frame.timerFun();//Open the timer

}
}
```