

5) Full Annotated Code

```
package spellinglearner;

import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;

//Checks which language to use then finds that text file and checks if the
word passed through is in the dictionary and returns true if it is and
false if it isn't
// Hbenbel. "Hbenbel/French-Dictionary." GitHub, 21 Nov. 2017,
github.com/hbenbel/French-Dictionary.
// "Spanish-English Downloadable Dictionary." Language Learning Stack
Exchange

class WordChecker {

    public static boolean check_for_word(String word, String language) {
        String filename = language + ".txt";
        try {
            BufferedReader in = new BufferedReader(new
FileReader(filename));
            String str;
            while ((str = in.readLine()) != null) {
                if (str.contains(word)) {
                    return true;
                }
            }
            in.close();
        } catch (IOException e) {
        }
        return false;
    }
}

package spellinglearner;
```

```

import java.io.Serializable;
import java.util.ArrayList;

public class nonWords implements Serializable {
    //creates a new arraylist of objects containing nonword
    (text,freq,text2,language)
    ArrayList<nonWord> nonWords = new ArrayList<>();

    public nonWord contains(String text) {
        for (int i = 0; i < nonWords.size(); i++) {
            if (nonWords.get(i).text.equals(text)) {
                return nonWords.get(i);
            }
        }
        return null;
    }
}

```

```

package spellinglearner;

```

```

import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.util.ArrayList;
import java.util.Scanner;
import java.util.logging.Level;
import java.util.logging.Logger;

```

```

/**
 *
 * @author mijitrenkel
 */

```

```

public class SpellingLearner {

    public static void main(String[] args) {
        new WelcomeFrame().setVisible(true);
    }
}

```

```

//Loops through wordlist (which is an arraylist of string) and uses the
wordchecker to see if it is in the dictionary if wordchecker returns false
it checks if it is already in the misspelledwords if yes then add one to

```

```

frequency if no then add word
    public static void test(String language, String text) {
        ArrayList<String> wordlist = new ArrayList();
        nonWords misspelledwords = new nonWords();

        String[] a = text.split(" ");
        for (String s : a) {
            wordlist.add(s);
        }

        //loop through wordlist, if not spelled correctly, add to
misspelledWords
        for (int i = 0; i < wordlist.size(); i++) {
            String word = wordlist.get(i);
            if (WordChecker.check_for_word(word, language) == false) {
                //add to misspelled words
                nonWord nonword = misspelledwords.contains(word);

                if (nonword == null) {
                    misspelledwords.nonWords.add(new nonWord(word, 1, "",
language));
                } else {
                    nonword.freq = nonword.freq + 1;
                }
            }
        }
        //Calling bubblesort to reorganise misspelledwords in terms of the
frequency
        BubbleSortExample.bubbleSort(misspelledwords);
        GUI myGUI = new GUI(misspelledwords, language);
        myGUI.setVisible(true);
    }
}

```

```

package spellinglearner;

```

```

import java.io.FileInputStream;
import java.io.FileNotFoundException;

```

```

import java.io.PrintStream;
import java.util.ArrayList;
import java.util.Scanner;
import java.util.logging.Level;
import java.util.logging.Logger;
import java.util.Arrays;
import java.util.Collections;

/**
 *
 * @author mijitrenkel
 */

public class BubbleSortExample {

    //Checks the value of the int freq and checks if it is greater than the
    previous one and swaps them until it is in the right order
    //“Java Program for Bubble Sort.” GeeksforGeeks, 12 Jan. 2018,
    www.geeksforgeeks.org/java-program-for-bubble-sort/.

    static void bubbleSort(nonWords list) {
        int n = list.nonWords.size();
        nonWord temp;
        for(int i=0; i < n; i++){
            for(int j=1; j < (n-i); j++){
                if(list.nonWords.get(j-1).freq <
list.nonWords.get(j).freq){
                    Collections.swap(list.nonWords, j, j-1);

                }

            }

        }

    }

}

package spellinglearner;

import java.io.Serializable;

```

```

public class nonWord implements Serializable {
    String text;
    int freq;
    String text2;
    String language;

    nonWord(String text, int freq, String text2, String language) {
        this.text = text;
        this.freq = 1;
        this.text2 = text2;
        this.language = language;
    }
}

```

```

package spellinglearner;

```

```

/**
 *
 * @author mijitrenkel
 */

```

```

public class WelcomeFrame extends javax.swing.JFrame {

```

```

    public WelcomeFrame() {
        initComponents();
    }

```

```

    @SuppressWarnings("unchecked")

```

```

    // <editor-fold defaultstate="collapsed" desc="Generated Code">

```

```

    private void initComponents() {

```

```

        language = new javax.swing.ButtonGroup();
        jPanel1 = new javax.swing.JPanel();
        spanishRadio = new javax.swing.JRadioButton();
        frenchRadio = new javax.swing.JRadioButton();
        englishRadio = new javax.swing.JRadioButton();
        jScrollPane1 = new javax.swing.JScrollPane();
        textArea = new javax.swing.JTextArea();

```

```

        goToTableButton = new javax.swing.JButton();
        checkButton = new javax.swing.JButton();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jPanel1.setBackground(new java.awt.Color(255, 255, 255));
        jPanel1.setBorder(javax.swing.BorderFactory.createMatteBorder(6, 6,
6, 6, new java.awt.Color(153, 204, 255)));
        jPanel1.setLayout(null);

        language.add(spanishRadio);
        spanishRadio.setFont(new java.awt.Font("Arial Hebrew", 1, 14)); //
NOI18N
        spanishRadio.setText("Spanish");
        jPanel1.add(spanishRadio);
        spanishRadio.setBounds(430, 100, 88, 23);

        language.add(frenchRadio);
        frenchRadio.setFont(new java.awt.Font("Arial Hebrew", 1, 14)); //
NOI18N
        frenchRadio.setText("French");
        jPanel1.add(frenchRadio);
        frenchRadio.setBounds(250, 100, 79, 23);

        language.add(englishRadio);
        englishRadio.setFont(new java.awt.Font("Arial Hebrew", 1, 14)); //
NOI18N
        englishRadio.setSelected(true);
        englishRadio.setText("English");
        jPanel1.add(englishRadio);
        englishRadio.setBounds(50, 100, 85, 23);

        textArea.setColumns(20);
        textArea.setFont(new java.awt.Font("Al Bayan", 0, 12)); // NOI18N
        textArea.setRows(5);
        jScrollPane1.setViewportView(textArea);

        jPanel1.add(jScrollPane1);

```

```

jScrollPane1.setBounds(30, 130, 510, 170);

goToTableButton.setFont(new java.awt.Font("Arial Hebrew", 1, 14));
// NOI18N
goToTableButton.setText("Table");
goToTableButton.addActionListener(new
java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        goToTableButtonActionPerformed(evt);
    }
});
jPanel1.add(goToTableButton);
goToTableButton.setBounds(360, 300, 84, 29);

checkButton.setBackground(new java.awt.Color(255, 153, 153));
checkButton.setFont(new java.awt.Font("Arial Hebrew", 1, 14)); //
NOI18N
checkButton.setText("Check ");
checkButton.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        checkButtonActionPerformed(evt);
    }
});
jPanel1.add(checkButton);
checkButton.setBounds(90, 300, 91, 29);

jLabel1.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/spellinglearner/fleeeg.jpg")
)); // NOI18N
jLabel1.setText("jLabel1");
jPanel1.add(jLabel1);
jLabel1.setBounds(40, 10, 119, 76);

jLabel2.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/spellinglearner/fleee.jpg"))
); // NOI18N
jPanel1.add(jLabel2);
jLabel2.setBounds(230, 10, 120, 84);

jLabel3.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/spellinglearner/fleeg2.jpg")
)); // NOI18N

```

```

jPanel1.add(jLabel3);
jLabel3.setBounds(410, 10, 120, 90);

javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
560, Short.MAX_VALUE)
    );
layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
349, Short.MAX_VALUE)
    );

pack();
} // </editor-fold>

private void checkButtonActionPerformed(java.awt.event.ActionEvent evt)
{

    String language = "english";
    if (frenchRadio.isSelected()) {
        language = "french";
    }
    if (spanishRadio.isSelected()) {
        language = "spanish";
    }
    String text = textArea.getText();
    SpellingLearner.test(language, text);
}

private void goToTableButtonActionPerformed(java.awt.event.ActionEvent
evt) {

    nonWords misspelledwords = new nonWords();
    new GUI(misspelledwords, "all").setVisible(true);
}

```



```

public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new WelcomeFrame().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton checkButton;
private javax.swing.JRadioButton englishRadio;
private javax.swing.JRadioButton frenchRadio;
private javax.swing.JButton goToTableButton;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JPanel jPanel1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.ButtonGroup language;
private javax.swing.JRadioButton spanishRadio;
private javax.swing.JTextArea textArea;
// End of variables declaration
}

/*
 * To change this license header, choose License Headers in Project
Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package spellinglearner;

import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.FileWriter;
import java.io.IOException;
import java.io.ObjectInputStream;
import java.io.ObjectOutputStream;
import java.util.logging.Level;

```

```

import java.util.logging.Logger;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;

/**
 *
 * @author mijitrenkel
 */
public class GUI extends javax.swing.JFrame {

    boolean error = false;

    nonWords misspelledwordGui;
    nonWords misspelledwords;

    public GUI(nonWords misspelledwords, String language) {
        initComponents();
        loadData();
        this.misspelledwords = misspelledwords;
        merge();
        updateTable();
    }

    private void updateTable() {
        DefaultTableModel tdm = (DefaultTableModel) tableList.getModel();

        for (int p = tdm.getRowCount() - 1; p >= 0; p--) {
            tdm.removeRow(p);
        }
        String language = "all";
        if (englishRadio.isSelected()) {
            language = "english";
        }
        if (spanishRadio.isSelected()) {
            language = "spanish";
        }
        if (frenchRadio.isSelected()) {
            language = "french";
        }

        for (nonWord word : this.misspelledwordGui.nonWords) {

```

```

        if (language.equals("all") || language.equals(word.language)) {
            tdm.addRow(new Object[]{word.text, word.freq, word.text2});
        }
    }

}

private void merge() {
    for (int i = 0; i < misspelledwords.nonWords.size(); i++) {
        String text = misspelledwords.nonWords.get(i).text;
        String language = misspelledwords.nonWords.get(i).language;

        boolean found = false;
        for (int j = 0; j < misspelledwordGui.nonWords.size(); j++) {
            if
(language.equals(misspelledwordGui.nonWords.get(j).language)) {
                if
(text.equals(misspelledwordGui.nonWords.get(j).text)) {
                    found = true;
                    misspelledwordGui.nonWords.get(j).freq +=
misspelledwords.nonWords.get(i).freq;
                }
            }
        }
        if (found == false) {

misspelledwordGui.nonWords.add(misspelledwords.nonWords.get(i));
        }

    }
}

private void saveCorrectSpellings() {

    DefaultTableModel tdm = (DefaultTableModel) tableList.getModel();
    for (int i = 0; i < misspelledwordGui.nonWords.size(); i++) {
        misspelledwordGui.nonWords.get(i).text2 = tdm.getValueAt(i,
2).toString();
    }
}

private void saveData() {

```

```

        for (int i = 0; i < misspelledwordGui.nonWords.size(); i++) {
            if (misspelledwordGui.nonWords.get(i).text2.equals("")) {
                error = true;
            } else {
                try {
                    FileOutputStream fos = new
FileOutputStream("misspelledwords.dat");
                    ObjectOutputStream oos = new ObjectOutputStream(fos);
                    oos.writeObject(misspelledwordGui);
                    fos.close();
                    oos.close();

                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        }
        if (error == true) {
            JOptionPane.showMessageDialog(this, "Please fill up each
column");
        }

    }

    private void loadData() {
        try {
            FileInputStream fis = new
FileInputStream("misspelledwords.dat");
            ObjectInputStream ois = new ObjectInputStream(fis);
            misspelledwordGui = (nonWords) ois.readObject();
            fis.close();
            ois.close();
        } catch (Exception e) {
            misspelledwordGui = new nonWords();
        }
    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

```

```

language = new javax.swing.ButtonGroup();
jScrollPane1 = new javax.swing.JScrollPane();
tableList = new javax.swing.JTable();
SaveButtonPage1 = new javax.swing.JButton();
learnbutton = new javax.swing.JButton();
deleteButton = new javax.swing.JButton();
editingField = new javax.swing.JTextField();
jLabel1 = new javax.swing.JLabel();
allRadio = new javax.swing.JRadioButton();
englishRadio = new javax.swing.JRadioButton();
frenchRadio = new javax.swing.JRadioButton();
spanishRadio = new javax.swing.JRadioButton();
cleartable = new javax.swing.JButton();
jLabel2 = new javax.swing.JLabel();
jLabel3 = new javax.swing.JLabel();

setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
getContentPane().setLayout(new
org.netbeans.lib.awtextra.AbsoluteLayout());

tableList.setFont(new java.awt.Font("Lucida Grande", 0, 13)); //
NOI18N
tableList.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {

        },
    new String [] {
        "Incorrect W", "Freq", "Correct W"
    }
) {
    Class[] types = new Class [] {
        java.lang.String.class, java.lang.Integer.class,
java.lang.String.class
    };
    boolean[] canEdit = new boolean [] {
        false, true, true
    };

    public Class getColumnClass(int columnIndex) {
        return types [columnIndex];
    }
}

```

```

        public boolean isCellEditable(int rowIndex, int columnIndex) {
            return canEdit [columnIndex];
        }
    });
    tableList.setShowGrid(false);
    jScrollPane1.setViewportViewView(tableList);
    if (tableList.getColumnModel().getColumnCount() > 0) {
        tableList.getColumnModel().getColumn(0).setResizable(false);
        tableList.getColumnModel().getColumn(1).setResizable(false);
    }

    getContentPane().add(jScrollPane1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(6, 0, 375, 391));

    SaveButtonPage1.setFont(new java.awt.Font("Lucida Grande", 1, 16));
// NOI18N
    SaveButtonPage1.setText("Save-Button");
    SaveButtonPage1.addActionListener(new
java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            SaveButtonPage1ActionPerformed(evt);
        }
    });
    getContentPane().add(SaveButtonPage1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(12, 397, -1, 40));

    learnbutton.setBackground(new java.awt.Color(0, 204, 153));
    learnbutton.setFont(new java.awt.Font("Lucida Grande", 1, 16)); //
NOI18N
    learnbutton.setText("Learn");
    learnbutton.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            learnbuttonActionPerformed(evt);
        }
    });
    getContentPane().add(learnbutton, new
org.netbeans.lib.awtextra.AbsoluteConstraints(190, 400, 100, 40));

    deleteButton.setFont(new java.awt.Font("Lucida Grande", 1, 15)); //
NOI18N
    deleteButton.setText("Delete");

```

```

        deleteButton.addActionListener(new java.awt.event.ActionListener()
{
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        deleteButtonActionPerformed(evt);
    }
});
getContentPane().add(deleteButton, new
org.netbeans.lib.awtextra.AbsoluteConstraints(470, 230, 84, -1));

        editingField.addActionListener(new java.awt.event.ActionListener()
{
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        editingFieldActionPerformed(evt);
    }
});
getContentPane().add(editingField, new
org.netbeans.lib.awtextra.AbsoluteConstraints(390, 170, 260, 53));

        jLabel1.setFont(new java.awt.Font("Lucida Grande", 1, 13)); //
NOI18N
        jLabel1.setText("Type the word you would like to delete");
        getContentPane().add(jLabel1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(390, 120, 270, 70));

        language.add(allRadio);
        allRadio.setFont(new java.awt.Font("Lucida Grande", 2, 14)); //
NOI18N
        allRadio.setSelected(true);
        allRadio.setText("All");
        allRadio.addItemListener(new java.awt.event.ItemListener() {
            public void itemStateChanged(java.awt.event.ItemEvent evt) {
                allRadioItemStateChanged(evt);
            }
        });
        getContentPane().add(allRadio, new
org.netbeans.lib.awtextra.AbsoluteConstraints(396, 17, 70, -1));

        language.add(englishRadio);
        englishRadio.setFont(new java.awt.Font("Lucida Grande", 2, 14)); //
NOI18N
        englishRadio.setText("English");
        englishRadio.addItemListener(new java.awt.event.ItemListener() {

```

```

        public void itemStateChanged(java.awt.event.ItemEvent evt) {
            englishRadioItemStateChanged(evt);
        }
    });
    getContentPane().add(englishRadio, new
org.netbeans.lib.awtextra.AbsoluteConstraints(396, 46, 100, -1));

    language.add(frenchRadio);
    frenchRadio.setFont(new java.awt.Font("Lucida Grande", 2, 14)); //
NOI18N
    frenchRadio.setText("French");
    frenchRadio.addItemListener(new java.awt.event.ItemListener() {
        public void itemStateChanged(java.awt.event.ItemEvent evt) {
            frenchRadioItemStateChanged(evt);
        }
    });
    getContentPane().add(frenchRadio, new
org.netbeans.lib.awtextra.AbsoluteConstraints(396, 75, 90, -1));

    language.add(spanishRadio);
    spanishRadio.setFont(new java.awt.Font("Lucida Grande", 2, 14)); //
NOI18N
    spanishRadio.setText("Spanish");
    spanishRadio.addItemListener(new java.awt.event.ItemListener() {
        public void itemStateChanged(java.awt.event.ItemEvent evt) {
            spanishRadioItemStateChanged(evt);
        }
    });
    getContentPane().add(spanishRadio, new
org.netbeans.lib.awtextra.AbsoluteConstraints(396, 104, 100, -1));

    cleartable.setFont(new java.awt.Font("Lucida Grande", 1, 16)); //
NOI18N
    cleartable.setText("Clear table");
    cleartable.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            cleartableActionPerformed(evt);
        }
    });
    getContentPane().add(cleartable, new
org.netbeans.lib.awtextra.AbsoluteConstraints(320, 400, 140, 40));

```



```
        jLabel2.setIcon(new  
javax.swing.ImageIcon(getClass().getResource("/spellinglearner/a43bb96e5f42  
e3569880488136f1b9fe.png"))); // NOI18N  
        getContentPane().add(jLabel2, new  
org.netbeans.lib.awtextra.AbsoluteConstraints(450, 180, 240, 300));
```

```
        jLabel3.setIcon(new  
javax.swing.ImageIcon(getClass().getResource("/spellinglearner/Screen Shot  
2019-11-07 at 12.50.12 AM.png"))); // NOI18N  
        jLabel3.setToolTipText("");  
        getContentPane().add(jLabel3, new  
org.netbeans.lib.awtextra.AbsoluteConstraints(0, 0, 660, 450));
```

```
        pack();  
    }// </editor-fold>
```

```
    private void SaveButtonPage1ActionPerformed(java.awt.event.ActionEvent  
evt) {  
        saveCorrectSpellings();  
        saveData();  
        merge();  
        loadData();  
    }
```

```
    private void learnbuttonActionPerformed(java.awt.event.ActionEvent evt)  
{  
        saveData();  
        if (error == false) {  
            new GameFrame().setVisible(true);  
        }  
  
    }
```

```
    private void editingFieldActionPerformed(java.awt.event.ActionEvent  
evt) {  
  
    }
```

```
    private void deleteButtonActionPerformed(java.awt.event.ActionEvent  
evt) {  
        String s = editingField.getText().trim();
```

```

        if (s.equals("")) {
            JOptionPane.showMessageDialog(this, "Cannot be blank");
            return;
        }
        boolean found = false;
        int i;
        for (i = 0; i < misspelledwordGui.nonWords.size(); i++) {
            if (s.equals(misspelledwordGui.nonWords.get(i).text)) {

                found = true;
                break;
            }
        }
        if (found) {
            misspelledwordGui.nonWords.remove(i);
            saveData();
            merge();
            loadData();
            updateTable();
            JOptionPane.showMessageDialog(this, "Removed Successfully");
        } else {
            JOptionPane.showMessageDialog(this, "Word not found");
        }
    }

    private void allRadioItemStateChanged(java.awt.event.ItemEvent evt) {

        updateTable();

    }

    private void englishRadioItemStateChanged(java.awt.event.ItemEvent evt)
    {

        updateTable();

    }

    private void frenchRadioItemStateChanged(java.awt.event.ItemEvent evt)
    {

        updateTable();

    }

```

```

        private void spanishRadioItemStateChanged(java.awt.event.ItemEvent evt)
        {

            updateTable();
        }

        private void cleartableActionPerformed(java.awt.event.ActionEvent evt)
        {
            try {

                java.io.FileWriter Writer = new
java.io.FileWriter("misspelledwords.dat");
                Writer.write("");
                Writer.close();
            } catch (IOException ex) {
                Logger.getLogger(GUI.class.getName()).log(Level.SEVERE, null,
ex);
            }
            misspelledwordGui = new nonWords();
            misspelledwords = new nonWords();
            updateTable();
        }

```

```

// Variables declaration - do not modify
private javax.swing.JButton SaveButtonPage1;
private javax.swing.JRadioButton allRadio;
private javax.swing.JButton cleartable;
private javax.swing.JButton deleteButton;
private javax.swing.JTextField editingField;
private javax.swing.JRadioButton englishRadio;
private javax.swing.JRadioButton frenchRadio;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.ButtonGroup language;
private javax.swing.JButton learnbutton;
private javax.swing.JRadioButton spanishRadio;
private javax.swing.JTable tableList;
// End of variables declaration

```

```
}
```

```
package spellinglearner;
```

```
import java.util.Locale;
```

```
import java.util.Scanner;
```

```
import javax.speech.Central;
```

```
import javax.speech.synthesis.Synthesizer;
```

```
import javax.speech.synthesis.SynthesizerModeDesc;
```

```
/**
```

```
 *
```

```
 * @author mijitrenkel
```

```
 */
```

```
public class TextToSpeech {
```

```
    static Synthesizer synthesizer;
```

```
    static {
```

```
        try {
```

```
            // set property as Kevin Dictionary
```

```
            System.setProperty("freetts.voices",
```

```
"com.sun.speech.freetts.en.us.cmu_us_kal.KevinVoiceDirectory");
```

```
            // Register Engine
```

```
Central.registerEngineCentral("com.sun.speech.freetts.jsapi.FreeTTSEngineCentral");
```

```
            // Create a Synthesizer
```

```
            synthesizer
```

```
                = Central.createSynthesizer(new
```

```
SynthesizerModeDesc(Locale.US));
```

```
            // Allocate synthesizer
```

```
            synthesizer.allocate();
```

```
            // Resume Synthesizer
```

```
            synthesizer.resume();
```

```

        // speaks the given text until queue is empty.
        // synthesizer.speakPlainText(s, null);
        // synthesizer.waitEngineState(Synthesizer.QUEUE_EMPTY);
        // Deallocate the Synthesizer.
        // synthesizer.deallocate();
    } catch (Exception e) {
        e.printStackTrace();
    }
}

public static void speak(String s) {

    try {
        synthesizer.speakPlainText(s, null);
        synthesizer.waitEngineState(Synthesizer.QUEUE_EMPTY);

    } catch (Exception e) {
        e.printStackTrace();
    }
}
}

```

```
package spellinglearner;
```

```

import java.io.FileInputStream;
import java.io.ObjectInputStream;
import java.util.ArrayList;
import java.util.Random;
import javax.swing.JOptionPane;

```

```

/**
 *
 * @author mijitrenkel
 */

```

```
public class GameFrame extends javax.swing.JFrame {
```

```

    nonWords misspelledwordGui;
    String text;
    ArrayList<String> belowavg = new ArrayList<String>();
    ArrayList<String> aboveavg = new ArrayList<String>();

```

```

int score = 0;
int outOf = 0;
ArrayList<String> words = new ArrayList();

public GameFrame() {
    initComponents();
    loadData();
    gameSetup();
    newGame();
}

private void loadData() {
    try {
        FileInputStream fis = new
FileInputStream("misspelledwords.dat");
        ObjectInputStream ois = new ObjectInputStream(fis);
        misspelledwordGui = (nonWords) ois.readObject();
        fis.close();
        ois.close();
    } catch (Exception e) {
        e.printStackTrace();
    }
}

private void gameSetup() {
    int sum = 0;
    for (int i = 0; i < misspelledwordGui.nonWords.size(); i++) {
        if (!misspelledwordGui.nonWords.get(i).text2.equals("")) {
            sum += misspelledwordGui.nonWords.get(i).freq;
        }
    }

    double avg = (double) sum / misspelledwordGui.nonWords.size();
    for (int i = 0; i < misspelledwordGui.nonWords.size(); i++) {
        if (!misspelledwordGui.nonWords.get(i).text2.equals("")) {
            if (misspelledwordGui.nonWords.get(i).freq < avg) {
                belowavg.add(misspelledwordGui.nonWords.get(i).text2);
            } else {
                aboveavg.add(misspelledwordGui.nonWords.get(i).text2);
            }
        }
    }
}

```

```

    }
}

private void newGame() {

    //0-20, 1-40, 2-60, 3-80, 4-100
    //0,1 words below avg 40%
    //2,3,4 words above avg 60%
    int r = new Random().nextInt(5);

    if (r < 2) {
        try {
            text = belowavg.get(new Random().nextInt(belowavg.size()));
        } catch (Exception e) {
            e.printStackTrace();
        }
    } else {
        text = aboveavg.get(new Random().nextInt(aboveavg.size()));
    }
    String str = "";
    for (int i = 0; i < text.length(); i++) {
        str += '\u2022';
        //u2022 = unicode value bullet
    }
    correctWordtext.setText(str);
    outOf++;
}

```

```

@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

```

```

    userInputWord = new javax.swing.JTextField();
    jLabel12 = new javax.swing.JLabel();
    jLabel11 = new javax.swing.JLabel();
    jLabel14 = new javax.swing.JLabel();
    soundButton = new javax.swing.JButton();
    checkButton = new javax.swing.JButton();
    correctWordtext = new javax.swing.JTextField();
    endgameButton = new javax.swing.JButton();

```

```

jLabel3 = new javax.swing.JLabel();
jLabel5 = new javax.swing.JLabel();

setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
setBackground(new java.awt.Color(238, 12, 238));
setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));
getContentPane().setLayout(new
org.netbeans.lib.awtextra.AbsoluteLayout());
getContentPane().add(userInputWord, new
org.netbeans.lib.awtextra.AbsoluteConstraints(20, 180, 230, 60));

jLabel2.setBackground(new java.awt.Color(255, 255, 255));
jLabel2.setFont(new java.awt.Font("Lucida Grande", 0, 15)); //
NOI18N
jLabel2.setText("Correct word here");
getContentPane().add(jLabel2, new
org.netbeans.lib.awtextra.AbsoluteConstraints(20, 20, 150, -1));

jLabel1.setFont(new java.awt.Font("Lucida Grande", 0, 15)); //
NOI18N
jLabel1.setText("User type word here");
getContentPane().add(jLabel1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(30, 160, 160, 20));

jLabel4.setBackground(new java.awt.Color(255, 255, 255));
jLabel4.setForeground(new java.awt.Color(255, 255, 255));
jLabel4.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/spellinglearner/Screen Shot
2019-11-06 at 11.20.58 PM.png"))); // NOI18N
jLabel4.setText("jLabel4");
getContentPane().add(jLabel4, new
org.netbeans.lib.awtextra.AbsoluteConstraints(20, 20, 180, 20));

soundButton.setBackground(new java.awt.Color(0, 204, 51));
soundButton.setFont(new java.awt.Font("Lucida Grande", 1, 15)); //
NOI18N
soundButton.setText("Play Sound");
soundButton.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        soundButtonActionPerformed(evt);
    }
}

```



```

    });
    getContentPane().add(soundButton, new
org.netbeans.lib.awtextra.AbsoluteConstraints(274, 41, -1, -1));

    checkButton.setFont(new java.awt.Font("Lucida Grande", 1, 15)); //
NOI18N
    checkButton.setText("Check");
    checkButton.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            checkButtonActionPerformed(evt);
        }
    });
    getContentPane().add(checkButton, new
org.netbeans.lib.awtextra.AbsoluteConstraints(270, 200, -1, -1));

    correctWordtext.setEditable(false);
    correctWordtext.setFont(new java.awt.Font("Lucida Grande", 0, 16));
// NOI18N
    getContentPane().add(correctWordtext, new
org.netbeans.lib.awtextra.AbsoluteConstraints(10, 40, 230, 38));

    endgameButton.setFont(new java.awt.Font("Lucida Grande", 1, 15));
// NOI18N
    endgameButton.setText("End Game");
    endgameButton.addActionListener(new java.awt.event.ActionListener()
{
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            endgameButtonActionPerformed(evt);
        }
    });
    getContentPane().add(endgameButton, new
org.netbeans.lib.awtextra.AbsoluteConstraints(30, 260, 120, -1));

    jLabel3.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/spellinglearner/4f484b7df1e8
69b567db021e99a8c3db.png"))); // NOI18N
    getContentPane().add(jLabel3, new
org.netbeans.lib.awtextra.AbsoluteConstraints(-30, 40, 450, 320));

    jLabel5.setBackground(new java.awt.Color(255, 255, 255));
    jLabel5.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/spellinglearner/Screen Shot

```

```

2019-11-07 at 12.50.12 AM.png")))); // NOI18N
    getContentPane().add(jLabel5, new
org.netbeans.lib.awtextra.AbsoluteConstraints(0, 0, 410, 320));

    pack();
} // </editor-fold>

private void soundButtonActionPerformed(java.awt.event.ActionEvent evt)
{
    TextToSpeech.speak(text);
}

private void checkButtonActionPerformed(java.awt.event.ActionEvent evt)
{
    String typedword = userInputWord.getText().trim();
    if (typedword.equals("")) {
        JOptionPane.showMessageDialog(this, "Please type the word");
    } else if (typedword.equalsIgnoreCase(text)) {
        JOptionPane.showMessageDialog(this, "Correct");
        score++;
        newGame();
    } else {
        JOptionPane.showMessageDialog(this, "Please type the correct
word displayed above");
        correctWordtext.setText(text);
        score--;
        words.add(text);
    }
    userInputWord.setText("");
}

private void endgameButtonActionPerformed(java.awt.event.ActionEvent
evt) {
    gameScoreGUI b = new gameScoreGUI(null, true);
    b.setData(score, outOf, words);
    b.setVisible(true);
    this.dispose();
}

public static void main(String args[]) {

```

```

        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new GameFrame().setVisible(true);
            }
        });
    }
}

```

```

// Variables declaration - do not modify
private javax.swing.JButton checkButton;
private javax.swing.JTextField correctWordtext;
private javax.swing.JButton endgameButton;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JButton soundButton;
private javax.swing.JTextField userInputWord;
// End of variables declaration
}

```

```

package spellinglearner;

```

```

import java.awt.Color;
import java.util.ArrayList;
import javax.swing.DefaultListModel;
import javax.swing.ImageIcon;
import javax.swing.ListModel;

```

```

/**

```

```

 *

```

```

 * @author mijitrenkel

```

```

 */

```

```

public class gameScoreGUI extends javax.swing.JDialog {

```

```

    public void setData(int score, int outOf, ArrayList<String> words) {
        this.score.setText(score + " / " + outOf);
        double percentage = score * 100.0 / outOf;
        this.percentage.setText(String.format("%.2f", percentage) + " % ");
        double stars = 0;
    }
}

```

```

if (percentage > 90) {
    stars = 5;
} else if (percentage > 80) {
    stars = 4.5;
} else if (percentage > 70) {
    stars = 4;
} else if (percentage > 60) {
    stars = 3.5;
} else if (percentage > 50) {
    stars = 3;
} else if (percentage > 40) {
    stars = 2.5;
} else if (percentage > 30) {
    stars = 2;
} else if (percentage > 20) {
    stars = 1.5;
} else if (percentage > 10) {
    stars = 1;
} else if (percentage > 0) {
    stars = 0.5;
} else {
    stars = 0;
}
String filename = "nostar.png";
l1.setIcon(new ImageIcon(filename));
l2.setIcon(new ImageIcon(filename));
l3.setIcon(new ImageIcon(filename));
l4.setIcon(new ImageIcon(filename));
l5.setIcon(new ImageIcon(filename));
filename = "fullstar.png";
switch ((int) stars) {
    case 5:
        l5.setIcon(new ImageIcon(filename));
    case 4:
        l4.setIcon(new ImageIcon(filename));
    case 3:
        l3.setIcon(new ImageIcon(filename));
    case 2:
        l2.setIcon(new ImageIcon(filename));
    case 1:
        l1.setIcon(new ImageIcon(filename));
}

```

```

    }
    filename = "halfstar.png";
    if (stars > (int) stars) {
        switch ((int) stars + 1) {
            case 5:
                l5.setIcon(new ImageIcon(filename));
                break;
            case 4:
                l4.setIcon(new ImageIcon(filename));
                break;
            case 3:
                l3.setIcon(new ImageIcon(filename));
                break;
            case 2:
                l2.setIcon(new ImageIcon(filename));
                break;
            case 1:
                l1.setIcon(new ImageIcon(filename));

        }
    }
    DefaultListModel<String> lm = new DefaultListModel();
    lm.removeAllElements();
    for (String s : words) {
        lm.addElement(s);
    }
    jList1.setModel(lm);
}

public gameScoreGUI(java.awt.Frame parent, boolean modal) {
    super(parent, modal);
    initComponents();
    this.getGlassPane().setBackground(Color.red);
}

@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

```

```

jPanel1 = new javax.swing.JPanel();
score = new javax.swing.JTextField();
jLabel8 = new javax.swing.JLabel();
jLabel11 = new javax.swing.JLabel();
percentage = new javax.swing.JTextField();
l5 = new javax.swing.JLabel();
l4 = new javax.swing.JLabel();
l3 = new javax.swing.JLabel();
l2 = new javax.swing.JLabel();
l1 = new javax.swing.JLabel();
jLabel12 = new javax.swing.JLabel();
jScrollPane1 = new javax.swing.JScrollPane();
jList1 = new javax.swing.JList<>();

setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
setBackground(new java.awt.Color(0, 255, 204));
setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));

jPanel1.setBackground(new java.awt.Color(255, 255, 255));
jPanel1.setBorder(new javax.swing.border.LineBorder(new
java.awt.Color(153, 153, 255), 3, true));
jPanel1.setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());

score.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        scoreActionPerformed(evt);
    }
});
jPanel1.add(score, new
org.netbeans.lib.awtextra.AbsoluteConstraints(64, 11, 100, 30));

jLabel8.setFont(new java.awt.Font("Lucida Grande", 1, 14)); //
NOI18N
jLabel8.setText("Score");
jPanel1.add(jLabel8, new
org.netbeans.lib.awtextra.AbsoluteConstraints(12, 20, 40, -1));

jLabel11.setFont(new java.awt.Font("Lucida Grande", 1, 14)); //
NOI18N
jLabel11.setText("Percentage");

```

```

        jPanel1.add(jLabel1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(218, 20, -1, -1));
        jPanel1.add(percentage, new
org.netbeans.lib.awtextra.AbsoluteConstraints(304, 11, 90, 30));

        l5.setText("        ");
        jPanel1.add(l5, new
org.netbeans.lib.awtextra.AbsoluteConstraints(260, 50, 50, 40));

        l4.setText("        ");
        jPanel1.add(l4, new
org.netbeans.lib.awtextra.AbsoluteConstraints(200, 50, 50, 40));

        l3.setText("        ");
        jPanel1.add(l3, new
org.netbeans.lib.awtextra.AbsoluteConstraints(140, 50, 50, 40));

        l2.setText("        ");
        jPanel1.add(l2, new
org.netbeans.lib.awtextra.AbsoluteConstraints(80, 50, 50, 40));

        l1.setText("        ");
        jPanel1.add(l1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(20, 50, 50, 40));

        jLabel2.setFont(new java.awt.Font("Arial Hebrew Scholar", 3, 14));
// NOI18N
        jLabel2.setText("Words got wrong");
        jPanel1.add(jLabel2, new
org.netbeans.lib.awtextra.AbsoluteConstraints(10, 100, 260, -1));

        jScrollPane1.setViewportView(jList1);

        jPanel1.add(jScrollPane1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(12, 123, 200, 190));

        javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

```

```

        .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
550, Short.MAX_VALUE)
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
346, Short.MAX_VALUE)
        );

        pack();
    }// </editor-fold>

    private void scoreActionPerformed(java.awt.event.ActionEvent evt) {

    }

    // Variables declaration - do not modify
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel8;
    private javax.swing.JList<String> jList1;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JLabel l1;
    private javax.swing.JLabel l2;
    private javax.swing.JLabel l3;
    private javax.swing.JLabel l4;
    private javax.swing.JLabel l5;
    private javax.swing.JTextField percentage;
    private javax.swing.JTextField score;
    // End of variables declaration
}

```