

guiAnagram

Alejandro Perez

Steps to run guiAnagram game:

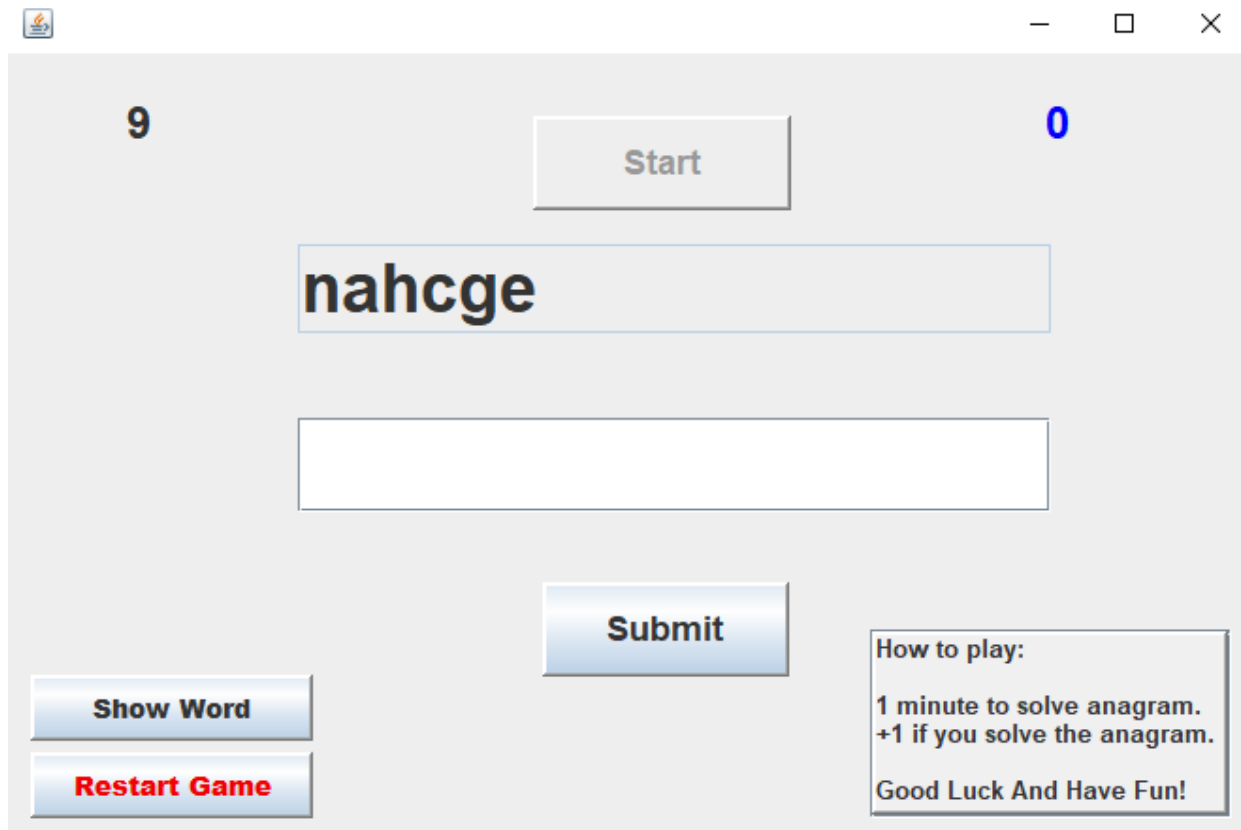
1. Download Java 8: <https://www.java.com/en/download/>
2. Download folder: https://drive.google.com/drive/folders/1Ai3eXNP4IK_EXc4r7xZOQM5rmfcD_Ccw?usp=sharing
3. Run guiAnagram.jar file.
4. Click on **Start**

****When running the *guiAnagram.jar* file, make sure *words.txt* is in the same folder.

Download Project:

- <https://github.com/alejandro1perez/guiAnagram.git>

UI?



Code?

```
Source Design History
1  /*
2   * To change this license header, choose License Headers in Project Properties.
3   * To change this template file, choose Tools | Templates
4   * and open the template in the editor.
5   */
6   package guianagram;
7
8   import java.awt.event.ActionEvent;
9   import java.awt.event.ActionListener;
10  import java.io.BufferedReader;
11  import java.io.File;
12  import java.io.FileReader;
13  import java.io.IOException;
14  import java.util.ArrayList;
15  import java.util.Collections;
16  import java.util.List;
17  import java.util.Random;
18  import javax.swing.Timer;
19
20  /**
21   *
22   * @author alejandro.perez
23   */
24  public class GuiFormAnagram extends javax.swing.JFrame {
25      static ArrayList<String> list = new ArrayList();
26      static Random rnd = new Random();
27      String original;
28      int counter = 0;
29      int count = 1;
30      int delay = 1000;
31      ActionListener al = new ActionListener() {
32          @Override
33          public void actionPerformed(ActionEvent e) {
34              txtTimer.setText(""+count++);
35              if (count == 60)
36              {
37                  t.stop();
38                  txtTimer.setText(""+0);
39                  txtShuffled.setText(original);
40                  txtScore.setText(""+ counter);
41                  txtWord.setText("");
42                  txtWord.setEnabled(false);
43                  btnSubmit.setEnabled(false);
44              }
45          }
46      };
47      Timer t = new Timer (delay,al);
48
49  }
```

```

49
50
51 //static String temp = "";
52 public static void LoadDictionary()
53 {
54     String s;
55     File f = new File("words.txt");
56     try
57     {
58         BufferedReader br = new BufferedReader(new FileReader(f));
59         while ((s = br.readLine()) != null)
60         {
61             list.add(s);
62         }
63     }
64     catch(IOException e)
65     {
66         System.out.println(e);
67     }
68 }
69 public static String randomWord(ArrayList<String> a)
70 {
71     return a.get(rnd.nextInt(a.size()));
72 }
73 public static String shuffleString(String s)
74 {
75     char[] c = s.toCharArray();
76     for(int i = 0; i < s.length()-1; i++)
77     {
78         int b = (char)(Math.random() * c.length);
79         char temp = c[i];
80         c[i] = c[b];
81         c[b] = temp;
82     }
83     return new String(c);
84 }
85
86
87 public GuiFormAnagram() {
88     initComponents();
89     LoadDictionary();
90 }
91
92 /**
93  * This method is called from within the constructor to initialize the form.
94  * WARNING: Do NOT modify this code. The content of this method is always
95  * regenerated by the Form Editor.
96  */
97 @SuppressWarnings("unchecked")

```

```

97 @SuppressWarnings("unchecked")
98 Generated Code
243
244 private void btnStartActionPerformed(java.awt.event.ActionEvent evt) {
245     original = randomWord(list);
246     String s = shuffleString(original);
247     txtShuffled.setText(s);
248     System.out.println(original);
249     t.start();
250     txtWord.setEnabled(true);
251     btnStart.setEnabled(false);
252     btnSubmit.setEnabled(true);
253     btnShow.setEnabled(true);
254     counter=0;
255     txtScore.setText("" + counter);
256 }
257
258
259 private void btnSubmitActionPerformed(java.awt.event.ActionEvent evt) {
260     String s = txtWord.getText();
261     if(s.equals(original))
262     {
263         counter++;
264         txtScore.setText("" + counter);
265         txtTimer.setText(""+0);
266         t.stop();
267         count=0;
268         t.start();
269         original = randomWord(list);
270         s = shuffleString(original);
271         txtShuffled.setText(s);
272         txtWord.setText("");
273         System.out.println(original);
274     }
275     else
276     {
277         txtScore.setText("Try Again!");
278         txtWord.setText("");
279     }
280     btnStart.setEnabled(false);
281 }
282
283
284 private void btnRestartActionPerformed(java.awt.event.ActionEvent evt) {
285     txtTimer.setText(""+0);
286     count = 0;
287     t.stop();
288     txtShuffled.setText("");
289     txtWord.setText("");

```

```

283
284 private void btnRestartActionPerformed(java.awt.event.ActionEvent evt) {
285     txtTimer.setText(""+0);
286     count = 0;
287     t.stop();
288     txtShuffled.setText("");
289     txtWord.setText("");
290     txtWord.setEnabled(false);
291     counter=1;
292     txtScore.setText("");
293     txtScore.setEnabled(true);
294     btnStart.setEnabled(true);
295     btnSubmit.setEnabled(false);
296     btnShow.setEnabled(false);
297
298 }
299
300 private void btnShowActionPerformed(java.awt.event.ActionEvent evt) {
301     txtShuffled.setText(original);
302     txtTimer.setText(""+0);
303     t.stop();
304     counter=1;
305     txtWord.setEnabled(false);
306     btnStart.setEnabled(false);
307     btnSubmit.setEnabled(false);
308
309 }
310
311
312 /**
313  * @param args the command line arguments
314  */
315 public static void main(String args[]) {
316     /* Set the Nimbus look and feel */
317     Look and feel setting code (optional)
318
319     /* Create and display the form */
320     java.awt.EventQueue.invokeLater(new Runnable() {
321         public void run() {
322             new GuiFormAnagram().setVisible(true);
323         }
324     });
325 }
326
327 // Variables declaration - do not modify
328 private javax.swing.JButton btnRestart;
329 private javax.swing.JButton btnShow;
330 private javax.swing.JButton btnStart;
331 private javax.swing.JButton btnSubmit;

```