

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

Folder src\main

7 printable files

(file list disabled)

src\main\java\org\example\INumberleModel.java

```
1 package org.example;
2
3 import java.util.ArrayList;
4
5 public interface INumberleModel{
6     int MAX_ATTEMPTS = 6;
7
8
9     void initialize();
10    boolean processInput(String input);
11    boolean isGameOver();
12    boolean isGameWon();
13
14    void setFlag3(boolean flag3);
15
16
17    void setFlag(boolean flag1);
18    void setTargetword(String targetword);
19    String getTargetWord();
20    StringBuilder getCurrentGuess();
21    int getRemainingAttempts();
22    void setRemainingAttempts(int val);
23    boolean Pr(String input);
24    boolean startNewGame();
25    String DefaultWord();
26
27    void remove( );
28
29 }
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

31
32
33

src/main/java/org/example/Junit.java

```
1 package org.example;
2
3 import org.junit.Test;
4
5 import org.junit.Before;
6
7
8
9 import static org.junit.Assert.*;
10
11
12 public class Junit {
13     private NumberleModel model;
14     @Before
15     public void Setup(){
16         model=new NumberleModel();
17     }
18     @Test
19     public void TestNumber(){
20         model.setTargetword("1+2=2+1");
21         assertEquals("1+2=2+1",model.getTargetWord());
22
23
24     }
25     @Test
26     public void ProcessInput(){
27         model.initialize();
28         assertFalse(model.processInput("1+2*3=7"));
29         assertFalse(model.isGameWon());
30     }
31     @Test
32     public void Remove() {
```

译 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

翻译成其他语言

不再提示

```
34     model.remove();
35     assertEquals(0, model.getCurrentGuess().length());
36 }
37 @Test
38 public void StartNewGame() {
39     model.initialize();
40     String oldWord = model.getTargetWord();
41     model.startNewGame();
42     assertNotEquals(oldWord, model.getTargetWord());
43 }
44 @Test
45 public void TestRemainingAttempts() {
46     model.initialize();
47     int initialAttempts = model.getRemainingAttempts();
48     model.setRemainingAttempts(initialAttempts - 1);
49     assertEquals(initialAttempts - 1, model.getRemainingAttempts());
50 }
51
52 @Test
53 public void Testing(){
54     model.initialize();
55     model.setTargetword("3+2=1+4");
56     assertTrue(model.processInput("3+2=1+4"));
57     assertTrue(model.gamewon);
58
59
60 }
61
62
63 }
64
65
```

src/main/java/org/example/Main.java

```
1 package org.example;
2
3
```

 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
5
6 public class Main {
7     public static void main(String[] args) {
8
9         INumberleModel model = new NumberleModel();
10        NumberleController controller = new NumberleController(model);
11        NumberleView view = new NumberleView(model, controller);
12
13
14    }
15 }
```

src/main/java/org/example/NumberleCLI.java

```
1 package org.example;
2
3 import java.awt.*;
4 import java.util.Scanner;
5 import net.objecthunter.exp4j.Expression;
6 import net.objecthunter.exp4j.ExpressionBuilder;
7
8
9
10 public class NumberleCLI {
11
12
13     private static final String ANSI_RESET = "\u001B[0m";
14     private static final String ANSI_GREEN = "\u001B[32m";
15     private static final String ANSI_YELLOW = "\u001B[33m";
16     private static final String ANSI_GRAY = "\u001B[90m";
17
18     private static final INumberleModel model = new NumberleModel();
19     private static final Scanner scanner = new Scanner(System.in);
20
21     public static void main(String[] args) {
22         model.startNewGame();
23         GAME();
24     }
```

译 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)[不再提示](#)

```
26 private static boolean correct(String input) {
27     String[] EQUAL = input.split("=");
28     if (EQUAL.length != 2) {
29         return false;
30     }
31     Expression expLeft = new ExpressionBuilder(EQUAL[0].trim()).build();
32     double leftResult = expLeft.evaluate();
33     Expression expRight = new ExpressionBuilder(EQUAL[1].trim()).build();
34     double rightResult = expRight.evaluate();
35     if (leftResult!=rightResult){
36         System.out.println("the left side is not equal right side ");
37     }
38     return Double.compare(leftResult, rightResult) == 0;
39 }
40 private static void GAME() {
41     while (!model.isGameOver()) {
42         KeyBoard();
43         Operator();
44         String input = Input();
45         if(input.length()<7){
46             System.out.println("your input length is must equal 7");
47         }
48         if (input.length()==7){
49             if (correct(input)) {
50                 boolean correct = model.Pr(input);
51
52                 COLOR(input, model.getTargetWord());
53                 if (correct) {
54                     break;
55                 }
56             } else {
57                 System.out.println("");
58             }
59         }
60     }
61
62     if (model.isGameWon()) {
63         System.out.println("you won the game!");
64     }
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)[不再提示](#)

```
67         System.out.println("The correct answer is " + model.getTargetWord());
68
69     }
70 }
71
72 private static void Operator() {
73     System.out.println("operator: + - * /");
74     System.out.println("number: 0 1 2 3 4 5 6 7 8 9");
75     System.out.println("other symbol: =");
76 }
77
78 private static String Input() {
79     System.out.print("please enter your guess: ");
80     return scanner.nextLine();
81 }
82 private static void KeyBoard() {
83     System.out.println("you have : " + model.getRemainingAttempts()+" "+"chances to try ");
84     System.out.println("your input guess is : " + model.getCurrentGuess());
85
86 }
87 private static void COLOR(String input, String Tarts) {
88     StringBuilder inputs = new StringBuilder();
89     inputs.setLength(0);
90     for (int i = 0; i < input.length(); i++) {
91         char c = input.charAt(i);
92         if (i < Tarts.length() && c == Tarts.charAt(i)) {
93             inputs.append(ANSI_GREEN).append(c);
94         } else if (Tarts.contains(String.valueOf(c))) {
95             inputs.append(ANSI_YELLOW).append(c);
96         } else {
97             inputs.append(ANSI_GRAY).append(c);
98         }
99     }
100     inputs.append(ANSI_RESET);
101     System.out.println(inputs.toString());
102 }
103 }
104
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
1 package org.example;
2
3 public class NumberleController {
4
5     private NumberleView view;
6     private INumberleModel model;
7
8     public NumberleController(INumberleModel model) {
9         this.model = model;
10    }
11    public int getRemainingAttempts() {
12        return model.getRemainingAttempts();
13    }
14    public void startNewGame() {
15        model.startNewGame();
16    }
17
18
19    public void SetFlag(Boolean Constant){
20
21        if (Constant==true){
22            model.setTargetword("5*1+2=7");
23        }
24
25    }
26    public void Flag2(Boolean flag){
27        model.setFlag(flag);
28        System.out.println("It's not correct answer,please try again and you have "+getRemainingAttempts()+" "+"chance ");
29
30    }
31    public void Flag3(Boolean Flag){
32        model.setFlag(Flag);
33        System.out.println(getTargetWord());
34    }
35    public void setView(NumberleView view) {
36        this.view = view;
37    }
38    public String getTargetWord() {
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
40     }
41     public void setRemainingAttempts(int val) {
42         model.setRemainingAttempts(val);
43     }
44     public void processInput(String input) {
45         model.processInput(input);
46     }
47     public boolean isGameOver() {
48         return model.isGameOver();
49     }
50     public boolean isGameWon() {
51         return model.isGameWon();
52     }
53 }
54 }
55 }
```

src/main/java/org/example/NumberleModel.java

```
1  package org.example;
2
3
4
5  import java.io.File;
6  import java.util.ArrayList;
7  import java.util.Random;
8  import java.util.Scanner;
9  import java.io.FileNotFoundException;
10
11 // NumberleModel.java
12 public class NumberleModel extends java.util.Observable implements INumberleModel{
13
14     private String TargetWord;
15     public ArrayList<String> NumberList=new ArrayList<>();
16     public int remainAttempts=6 ;
17     public boolean gamewon;
18     public boolean Flag1=false;
19     public boolean Flag2=false ;
```


译 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
21
22     public String DefaultNumberal="5*1+2=7";
23
24     private StringBuilder CurrentGuess;
25
26
27
28     public void initialize(){
29
30
31         this.LoadList();
32         Random random=new Random();
33         int index =random.nextInt(NumberList.size());
34         if (Flag1==true){
35             System.out.println("this is not ");
36         }
37         if (Flag2==true){
38             System.out.println(getTargetWord());
39         }
40         if (Flag3==true){
41             TargetWord=DefaultNumberal;
42         }else {
43             TargetWord=NumberList.get(index);
44         }
45         CurrentGuess=new StringBuilder(TargetWord.length());
46         remainAttempts=MAX_ATTEMPTS;
47         gamewon=false;
48         setChanged();
49         notifyObservers();
50
51
52
53
54     }
55     public boolean ProcessInput(String input){
56
57
58         if (input.length() < 7){
59             return false;
```

译 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

翻译成其他语言

不再提示

```
62
63     if (CurrentGuess.toString().equals(TargetWord)) {
64         gamewon = true;
65     }
66     setChanged();
67     notifyObservers();
68     return gamewon;
69 }
70
71 public void remove(){
72     if (CurrentGuess.length() > 0){
73         CurrentGuess.deleteCharAt(CurrentGuess.length() - 1);
74         setChanged();
75         notifyObservers();
76     }
77 }
78
79
80 public void LoadList(){
81     try {
82         File file = new File("C:/Users/93678/IdeaProjects/Numbera/src/equations.txt");
83         assert file.exists(): "File should be exists";
84         Scanner scanner = new Scanner(file);
85
86         while(scanner.hasNextLine()){
87             this.NumberList.add(scanner.nextLine());
88
89         }
90         scanner.close();
91
92     } catch (FileNotFoundException e) {
93         System.out.println("This File is can't read ");
94     }
95
96 }
97
98
99 @Override
100 public boolean processInput(String input) {
```

译 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
103
104     @Override
105     public boolean isGameOver() {
106         assert remainAttempts >= 0 : "remainAttempts should be non-negative";
107         return remainAttempts <= 0 || gamewon ;
108     }
109
110     @Override
111     public boolean isGameWon() {
112         return gamewon;
113     }
114
115
116
117     @Override
118     public void setFlag(boolean flag2) {
119         this.Flag2=flag2;
120     }
121     @Override
122     public void setFlag3(boolean flag3) {
123         this.Flag3=flag3;
124     }
125
126
127
128
129     @Override
130     public void setTargetword(String targetword) {
131         assert targetword != null : "targetword should not be null";
132         assert targetword.length() >= 7 : "targetword length should be >= 7";
133         this.TargetWord=targetword;
134     }
135
136     @Override
137     public String getTargetWord() {
138         return TargetWord;
139     }
140
141     @Override
```

译 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
144     }
145
146     @Override
147     public int getRemainingAttempts() {
148         assert remainAttempts >= 0 : "remainAttempts should be non-negative";
149         return remainAttempts;
150     }
151
152     @Override
153     public void setRemainingAttempts(int attempt) {
154         assert attempt >= 0 : "val should be non-negative";
155         remainAttempts = attempt;
156     }
157
158
159     @Override
160     public boolean Pr(String input) {
161         assert input.length() <= 7 : "Input length should be <= 7";
162         if (input.length() < 7){
163             System.out.println("input length must =7");
164         }
165         CurrentGuess = new StringBuilder(input);
166
167         if (CurrentGuess.toString().equals(TargetWord)) {
168             gamewon = true;
169         }else {
170             remainAttempts--;
171         }
172     }
173     return gamewon;
174 }
175
176
177 @Override
178 public boolean startNewGame() {
179     initialize();
180     return false;
181 }
182
183
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
185         return null;
186     }
187
188
189
190
191
192
193
194
195
196     }
197
```

src/main/java/org/example/Number1eView.java

```
1  package org.example;
2
3
4  import javax.swing.*;
5
6  import java.awt.*;
7  import java.awt.event.ActionEvent;
8  import java.awt.event.ActionListener;
9
10 import java.util.Observable;
11 import java.util.Observer;
12
13
14 import net.objecthunter.exp4j.Expression;
15 import net.objecthunter.exp4j.ExpressionBuilder;
16
17
18
19
20 public class Number1eView extends JFrame implements Observer {
21     private JPanel Panel;
22
```

译 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
24 private final INumberleModel model;
25 public JButton button[][]=new JButton[6][7];
26 private JTextField TextField;
27 private int currentRow=0;
28 public static final int ROWS = 6;
29 public static final int COLS = 7;
30 public static final int MAX_INPUT_LENGTH = 7;
31 private static final Color CORRECT_COLOR = Color.GREEN;
32 //The entered text contains letters but in the wrong order
33 private static final Color INCORRECT_COLOR = Color.YELLOW;
34 private final Color defaultColor = UIManager.getColor("Button.background");
35 //
36 private JPanel row1;
37 private JPanel row2;
38
39
40 public NumberleView(INumberleModel model,NumberleController controller){
41     this.controller=controller;
42     this.model=model;
43     controller.startNewGame();
44     Initialize();
45     showGameStartReminder();
46     ((NumberleModel)this.model).addObserver(this);
47     this.controller.setView(this);
48     update((NumberleModel)this.model, null);
49
50
51
52 }
53 public void Initialize(){
54
55     try {
56         UIManager.setLookAndFeel("com.sun.java.swing.plaf.windows.WindowsClassicLookAndFeel");
57     } catch (Exception e) {
58         e.printStackTrace();
59     }
60
61     setTitle("Numberle");
62     setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
65     setSize(800, 800);
66     setLayout(new BorderLayout());
67     Panel = new JPanel(new GridLayout(6, 7)); // Use GridLayout for buttons
68
69
70
71
72
73
74     setLocationRelativeTo(null);
75     JPanel inputPanel=new JPanel();
76     TextField=new JTextField(7);
77     inputPanel.add(TextField);
78     add(inputPanel,BorderLayout.NORTH);
79     inputPanel.setVisible(false);
80     String input=TextField.getText();
81
82     assert (input)!=null;
83
84     JPanel keyBoardPanel = new JPanel();
85     keyBoardPanel.setLayout(new BoxLayout(keyBoardPanel, BoxLayout.Y_AXIS));
86     row1=new JPanel();
87     row1.setLayout(new FlowLayout());
88     for (int i = 1; i <= 9; i++) {
89         JButton button = new JButton(Integer.toString(i));
90         button.setFont(new Font("Serif", Font.PLAIN, 30));
91         button.addActionListener(new ButtonClickListener());
92         row1.add(button);
93     }
94     JButton zeroButton = new JButton("0");
95     zeroButton.setFont(new Font("Serif", Font.PLAIN, 30));
96     zeroButton.addActionListener(new ButtonClickListener());
97     row1.add(zeroButton);
98
99     row2 = new JPanel();
100    row2.setLayout(new FlowLayout());
101    String[] operators = {"Delete", "+", "-", "*", "/", "=", "Enter"};
102    for (String operator : operators) {
103        JButton button = new JButton(operator);
104        button.setFont(new Font("Serif", Font.PLAIN, 30));
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
106         row2.add(button);
107     }
108     keyBoardPanel.add(row1);
109     keyBoardPanel.add(row2);
110
111     add(keyBoardPanel, BorderLayout.SOUTH);
112
113     JPanel ButtonPanel=new JPanel();
114
115
116
117     JButton yoursetting=new JButton("Constant_Word");
118
119
120     JButton target=new JButton("Target");
121
122     yoursetting.addActionListener(e -> {
123         Fixed();
124     });
125     target.addActionListener(e -> {
126         target();
127     });
128
129
130
131
132
133
134     ButtonPanel.add(target);
135     ButtonPanel.add(yoursetting);
136     add(ButtonPanel, BorderLayout.NORTH);
137
138
139
140     for (int i=0;i<6;i++){
141         for (int j=0;j<7;j++){
142             button[i][j]=new JButton("");
143             button[i][j].setPreferredSize(new Dimension(100,100));
144             Panel.add(button[i][j]);
145         }
146     }
```


[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
147         Panel.setVisible(true);
148
149     }
150
151
152
153
154     setVisible(true);
155     change();
156
157 }
158 public void Fixed(){
159     controller.SetFlag(true);
160     System.out.println(controller.getTargetWord());
161 }
162
163 public void change() {
164     String input = TextField.getText();
165     updateButtonsFromInput(input);
166 }
167
168
169 public void target(){
170     controller.Flag3(true);
171 }
172
173
174 private class ButtonClickListener implements ActionListener {
175     @Override
176     public void actionPerformed(ActionEvent e) {
177         JButton source = (JButton) e.getSource();
178         String buttonText = source.getText();
179         switch (buttonText) {
180             case "Remove":
181                 clearDisplay();
182                 break;
183             case "Enter":
184                 Enter();
185                 break;
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)[不再提示](#)

```
188         updateDisplay(buttonText);
189         updateButtonsFromInput(TextField.getText());
190     }
191     controller.isGameOver();
192 }
193 }
194 }
195 private void TargetColor(char[] target, String input) {
196     char[] userInput = input.toCharArray();
197
198     for (int col = 0; col < 7; col++) {
199
200         char targetChar = target[col];
201         char inputChar = userInput[col];
202
203         if (targetChar == inputChar) {
204             button[currentRow][col].setBackground(Color.GREEN);
205         } else if (new String(target).indexOf(inputChar) != -1) {
206             button[currentRow][col].setBackground(Color.ORANGE);
207         } else {
208             button[currentRow][col].setBackground(Color.GRAY);
209         }
210
211         button[currentRow][col].setText(String.valueOf(inputChar));
212
213
214         String buttonText = button[currentRow][col].getText();
215
216
217         for (Component component : row1.getComponents()) {
218             if (component instanceof JButton) {
219                 JButton keyboardButton = (JButton) component;
220                 if (keyboardButton.getText().equals(buttonText)) {
221                     if (targetChar == inputChar) {
222                         keyboardButton.setBackground(Color.GREEN);
223                     } else if (new String(target).indexOf(inputChar) != -1) {
224                         keyboardButton.setBackground(Color.ORANGE);
225                     } else {
226                         keyboardButton.setBackground(Color.GRAY);
227                     }
228                 }
229             }
230         }
231     }
232 }
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
229         }
230     }
231 }
232
233
234     for (Component component : row2.getComponents()) {
235         if (component instanceof JButton) {
236             JButton keyboardButton = (JButton) component;
237             if (keyboardButton.getText().equals(buttonText)) {
238                 if (targetChar == inputChar) {
239                     keyboardButton.setBackground(Color.GREEN);
240                 } else if (new String(target).indexOf(inputChar) != -1) {
241                     keyboardButton.setBackground(Color.ORANGE);
242                 } else {
243                     keyboardButton.setBackground(Color.GRAY);
244                 }
245                 break;
246             }
247         }
248     }
249 }
250 }
251
252
253
254 private void updateButtonsFromInput(String input) {
255     int rowIndex = currentRow;
256
257     int colIndex = 0;
258     char[] chars = input.toCharArray();
259
260     for (char c : chars) {
261         if (Character.isDigit(c) || "+-*/=".indexOf(c) != -1) {
262             if (colIndex < 7) {
263                 button[rowIndex][colIndex].setText(String.valueOf(c));
264                 colIndex++;
265             }
266         }
267     }
268 }
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
270     }
271
272
273 }
274
275 public void Enter() {
276     String input = TextField.getText();
277
278     // Check if input length is less than 7
279     if (input.length() < 7) {
280         controller.Flag2(true);
281         TooShort();
282
283         return; // Return to avoid further processing
284     }
285
286     boolean containsOperator = false; // Flag to check if an operator is present
287
288     for (char c : input.toCharArray()) {
289         if ("+-*/".indexOf(c) != -1) {
290             containsOperator = true; // Set flag if an operator is found
291             break;
292         }
293     }
294
295     if (input.length() == 7) {
296         boolean containsEquals = input.contains("=");
297         if (!containsEquals) {
298
299             NoEqualSign();
300         } else if (!containsOperator) {
301             Symbol();
302         } else {
303             boolean isCorrect = validateEquation(input);
304             if (isCorrect) {
305                 char[] target = controller.getTargetWord().toCharArray();
306                 validateEquation(input);
307
308                 controller.setRemainingAttempts(controller.getRemainingAttempts() - 1);
309             }
310         }
311     }
312 }
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
311         TargetColor(target,input);
312         currentRow++;
313         if (currentRow>5){
314             currentRow=0;
315         }
316         controller.processInput(input);
317         TextField.setText(""); // Clear the input text field
318         if (controller.getRemainingAttempts()==0){
319             LOSEGame();
320         }
321     }
322     else{
323
324         NotEqual();
325     }
326 }
327 }
328
329 }
330
331
332
333
334
335
336 private boolean validateEquation(String input) {
337     String[] parts = input.split("=");
338     if (parts.length != 2) {
339         return false;
340     }
341
342     Expression expLeft = new ExpressionBuilder(parts[0].trim()).build();
343     double leftResult = expLeft.evaluate();
344     Expression expRight = new ExpressionBuilder(parts[1].trim()).build();
345     double rightResult = expRight.evaluate();
346
347     return Double.compare(leftResult, rightResult) == 0;
348 }
349
350 }
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
352
353
354
355     public void clearDisplay(){
356         String currentText=TextField.getText();
357         if (!currentText.isEmpty()){
358             String newText=currentText.substring(0,currentText.length()-1);
359             TextField.setText(newText);
360             updateButtonsFromInput(newText);
361
362         }
363     }
364
365
366 }
367
368
369
370 public void updateDisplay(String text){
371     if (TextField.getText().length()+text.length()>7){
372         text = text.substring(0, 7 - TextField.getText().length());
373     }
374     TextField.setText(TextField.getText()+text);
375
376 }
377
378
379
380 public void TooShort(){
381     JFrame reminderFrame = new JFrame("reminder");
382     reminderFrame.setBounds(625,300, 400, 200);
383     reminderFrame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
384     JLabel reminderLabel = new JLabel("Too Short !");
385     reminderLabel.setFont(new Font("Serif", Font.BOLD, 25));
386     reminderLabel.setHorizontalAlignment(JLabel.CENTER);
387
388     JPanel reminderPanel = new JPanel(new BorderLayout());
389     reminderPanel.add(reminderLabel, BorderLayout.CENTER);
390     reminderFrame.getContentPane().add(reminderPanel);
391 }
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
393
394     reminderFrame.setVisible(true);
395
396     Timer timer = new Timer(2000, e -> {
397         reminderFrame.dispose();
398     });
399     timer.setRepeats(false);
400     timer.start();
401
402
403
404 }
405 public void Symbol(){
406     JFrame reminderFrame = new JFrame("reminder");
407     reminderFrame.setBounds(625,300, 400, 200);
408     reminderFrame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
409     JLabel reminderLabel = new JLabel("There must be at least one sign +-*/ !");
410     reminderLabel.setFont(new Font("Serif", Font.BOLD, 25));
411     reminderLabel.setHorizontalAlignment(JLabel.CENTER);
412
413     JPanel reminderPanel = new JPanel(new BorderLayout());
414     reminderPanel.add(reminderLabel, BorderLayout.CENTER);
415     reminderFrame.getContentPane().add(reminderPanel);
416
417
418
419     reminderFrame.setVisible(true);
420
421     Timer timer = new Timer(2000, e -> {
422         reminderFrame.dispose();
423     });
424     timer.setRepeats(false);
425     timer.start();
426
427
428 }
429
430
431 public void NotEqual(){
432     JFrame reminderFrame = new JFrame("reminder");
433     reminderFrame.setBounds(625,300, 400, 200);
434     reminderFrame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
435     JLabel reminderLabel = new JLabel("There must be at least one sign +-*/ !");
436     reminderLabel.setFont(new Font("Serif", Font.BOLD, 25));
437     reminderLabel.setHorizontalAlignment(JLabel.CENTER);
438
439     JPanel reminderPanel = new JPanel(new BorderLayout());
440     reminderPanel.add(reminderLabel, BorderLayout.CENTER);
441     reminderFrame.getContentPane().add(reminderPanel);
442
443
444
445     reminderFrame.setVisible(true);
446
447     Timer timer = new Timer(2000, e -> {
448         reminderFrame.dispose();
449     });
450     timer.setRepeats(false);
451     timer.start();
452
453
454 }
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)[不再提示](#)

```
434 reminderFrame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
435 JLabel reminderLabel = new JLabel("the left side is not equal to the right");
436 reminderLabel.setFont(new Font("Serif", Font.BOLD, 25));
437 reminderLabel.setHorizontalAlignment(JLabel.CENTER);
438
439 JPanel reminderPanel = new JPanel(new BorderLayout());
440 reminderPanel.add(reminderLabel, BorderLayout.CENTER);
441 reminderFrame.getContentPane().add(reminderPanel);
442
443
444
445 reminderFrame.setVisible(true);
446
447 Timer timer = new Timer(2000, e -> {
448     reminderFrame.dispose();
449 });
450 timer.setRepeats(false);
451 timer.start();
452
453 }
454 private void showGameStartReminder() {
455     JFrame reminderFrame = new JFrame("reminder");
456     reminderFrame.setBounds(625, 300, 400, 200);
457     reminderFrame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
458     JLabel reminderLabel = new JLabel("Guess the first equations! ");
459     reminderLabel.setFont(new Font("Serif", Font.BOLD, 25));
460     reminderLabel.setHorizontalAlignment(JLabel.CENTER);
461
462     JPanel reminderPanel = new JPanel(new BorderLayout());
463     reminderPanel.add(reminderLabel, BorderLayout.CENTER);
464     reminderFrame.getContentPane().add(reminderPanel);
465
466
467
468     reminderFrame.setVisible(true);
469
470     Timer timer = new Timer(2000, e -> {
471         reminderFrame.dispose();
472     });
```


[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
475     }
476
477     public void NoEqualSign(){
478         JFrame reminderFrame = new JFrame("reminder");
479         reminderFrame.setBounds(625,300, 400, 200);
480         reminderFrame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
481         JLabel reminderLabel = new JLabel("No equal = sign !");
482         reminderLabel.setFont(new Font("Serif", Font.BOLD, 25));
483         reminderLabel.setHorizontalAlignment(JLabel.CENTER);
484
485         JPanel reminderPanel = new JPanel(new BorderLayout());
486         reminderPanel.add(reminderLabel, BorderLayout.CENTER);
487         reminderFrame.getContentPane().add(reminderPanel);
488
489
490
491         reminderFrame.setVisible(true);
492
493         Timer timer = new Timer(2000, e -> {
494             reminderFrame.dispose();
495         });
496         timer.setRepeats(false);
497         timer.start();
498     }
499     public void LOSEGame(){
500         JFrame reminderFrame = new JFrame("reminder");
501         reminderFrame.setBounds(625,300, 400, 200);
502         reminderFrame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
503         Button Restart=new Button("Replay");
504         Restart.addActionListener(e -> {
505
506             controller.startNewGame();
507             currentRow=0;
508
509
510
511             for (int row = 0; row < ROWS; row++) {
512                 for (int col = 0; col < COLS; col++) {
513                     button[row][col].setText("");
514                 }
515             }
516             Restart.setEnabled(true);
517         });
518     }
519 }
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
516     }
517
518     for (Component component : row1.getComponents()) {
519         if (component instanceof JButton) {
520             component.setEnabled(true);
521             component.setBackground(defaultColor);
522         }
523     }
524     for (Component component : row2.getComponents()) {
525         if (component instanceof JButton) {
526             component.setEnabled(true);
527             component.setBackground(defaultColor);
528         }
529     }
530     reminderFrame.dispose();
531
532 });
533 Restart.setBounds(10,20,10,10);
534
535 JLabel reminderLabel = new JLabel("You lost!");
536 reminderLabel.setFont(new Font("Serif", Font.BOLD, 25));
537
538 reminderLabel.setHorizontalAlignment(JLabel.CENTER);
539
540 JPanel reminderPanel = new JPanel(new BorderLayout());
541 reminderPanel.add(reminderLabel, BorderLayout.CENTER);
542 reminderPanel.add(Restart,BorderLayout.NORTH);
543
544 reminderFrame.getContentPane().add(reminderPanel);
545
546 Restart.setVisible(true);
547 reminderFrame.setVisible(true);
548
549 }
550
551 public void WonGame(){
552     JFrame reminderFrame = new JFrame("reminder");
553     reminderFrame.setBounds(625,300, 400, 200);
554     reminderFrame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
555     Restart.setBounds(10,20,10,10);
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)[不再提示](#)

```
557         currentRow=0;
558         controller.startNewGame();
559
560
561         for (int row = 0; row < ROWS; row++) {
562             for (int col = 0; col < COLS; col++) {
563                 button[row][col].setText("");
564                 button[row][col].setBackground(defaultColor);
565             }
566         }
567
568         for (Component component : row1.getComponents()) {
569             if (component instanceof JButton) {
570                 component.setEnabled(true);
571             };
572             component.setBackground(defaultColor);
573         }
574     }
575     for (Component component : row2.getComponents()) {
576         if (component instanceof JButton) {
577             component.setEnabled(true);
578             component.setBackground(defaultColor);
579         }
580     }
581     reminderFrame.dispose();
582
583 });
584 Restart.setBounds(10,20,10,10);
585
586 JLabel reminderLabel = new JLabel("You won!");
587 reminderLabel.setFont(new Font("Serif", Font.BOLD, 25));
588
589 reminderLabel.setHorizontalAlignment(JLabel.CENTER);
590
591 JPanel reminderPanel = new JPanel(new BorderLayout());
592 reminderPanel.add(reminderLabel, BorderLayout.CENTER);
593 reminderPanel.add(Restart,BorderLayout.NORTH);
594
595 reminderFrame.getContentPane().add(reminderPanel);
```

[译](#) 极速翻译 检测到当前网页不是中文网页，是否要翻译成中文？

翻译

不翻译

[翻译成其他语言](#)

[不再提示](#)

```
598 Restart.setVisible(true);
599 reminderFrame.setVisible(true);
600
601 }
602 @Override
603
604 public void update(Observable o, Object arg) {
605     controller.getRemainingAttempts();
606
607     if (controller.isGameOver()) {
608         if (controller.isGameWon()) {
609
610             WonGame();
611         } else if (controller.getRemainingAttempts() == 0) {
612             LOSEGame();
613         }
614
615         for (Component component : row1.getComponents()) {
616             if (component instanceof JButton) {
617                 component.setEnabled(false);
618             }
619         }
620         for (Component component : row2.getComponents()) {
621             if (component instanceof JButton) {
622                 component.setEnabled(false);
623             }
624         }
625     }
626 }
627 }
628 }
```