Lab Guessing Game

GameConsole.java | GameDialog.java | GameSolver.java | Main.java | MainSover.j

```
2  * To change this license header, choose License Headers in Project Proper
6  package guessinggame;
7
8  import java.util.Scanner;
9
10 public class GameConsole {
11     public int play(NumberGame game) {
12         Scanner console = new Scanner(System.in);
13                 GameSolver gameSolver = new GameSolver();
14         int guess =0;
15                 int answer;
16                 boolean correct = false;
17         System.out.println( game.toString() );
18
19         System.out.println( game.getMessage() );
20
21         while(correct == false){
22             System.out.print("Your answer? ");
23                     answer = gameSolver.play(game);
24                     System.out.println(answer);
25                     guess = answer;
26             correct = ((MyGameNutpapat)(game)).guess(answer);
27
28             System.out.println( game.getMessage() );
29                     }
30             return guess;
31     }
32
33 }
```

GameSolver.java | Main.java | MainSover.java | MyGameNutpapat.java

```
2  * To change this license header, choose License Headers in Pr
6  package guessinggame;
7  import java.util.Random;
8  public class GameSolver {
9      public int play(NumberGame game){
10         long seed = System.nanoTime();
11         Random rand = new Random(seed);
12         return rand.nextInt(game.getUpperBound())+1;
13     }
14
15 }
16
```

MainSover.java | MyGameNutpapat.java | NumberGame.java

```
1  package guessinggame;
2  import java.util.Random;
3  public class MyGameNutpapat extends NumberGame{
4      private int secret, upperBound;
5      public MyGameNutpapat(int upperBound){
6          this.upperBound = upperBound;
7           long seed = System.nanoTime();
8              Random rand = new Random(seed);
9           this.secret = rand.nextInt(upperBound)+1;
10          super.setMessage( "I'm thinking of a number between 1 and "+upperBound);
11     }
12      public boolean guess(int number) {
13         if (number == secret) {
14             setMessage("Correct! The answer is "+secret);
15                 this.count++;
16             return true;
17         }
18         if (number < secret) {
19             setMessage("Your answer is too small.");
20         }
21         else  {
22             setMessage("Your answer is too large.");
23         }
24         this.count++;
25         return false;
26     }
27      /** Get the game upper bound. */
28     public int getUpperBound() {
29         return upperBound;
30     }
31         public void setUpperBound(int upperBound){
32             this.upperBound = upperBound;
33         }
34     public String showSecret() {
35         return "secret is "+secret;
36     }
37     @Override
38     public String toString() {
39         return "Guess a secret number between 1 and 100";
40     }
41     public int getCount(){
42         return this.count;
43     }
```

GameDialog.java | GameSolver.java | Main.java | MainSover.java | MyGameNutpapat.java | NumberGame.java

```
2  * To change this license header, choose License Headers in Project Properties.
6  package guessinggame;
7
8  import java.util.Scanner;
10 public class GameDialog {
11     public int play(NumberGame game) {
12 //      Scanner console = new Scanner(System.in);
13             GameSolver gameSolver = new GameSolver();
14         int guess =0;
15                 int answer;
16                 boolean correct;
17         // describe the game
18         System.out.println( game.toString() );
19                 JOptionPane.showMessageDialog(null, game.toString(), "GuessingGame",  JOptionPane.INFORMATION_MESSAGE);
20
21         System.out.println( game.getMessage() );
22                 JOptionPane.showMessageDialog(null, game.getMessage(), "GuessingGame",  JOptionPane.INFORMATION_MESSAGE);
23                System.out.println(((MyGameNutpapat)(game)).showSecret());
24                 do{
25                 guess = Integer.parseInt(JOptionPane.showInputDialog(null, "Your answer?"));
26                 correct = ((MyGameNutpapat)(game)).guess(guess);
27                 JOptionPane.showMessageDialog(null, game.getMessage(), "Result",JOptionPane.WARNING_MESSAGE);
28                 if(correct) {
29                     int playAgain = JOptionPane.showConfirmDialog(null, "Want to play again?", " Question", JOptionPane.YES_NO_OPTION)
30                     // yes =0,  no =1
31                     if(playAgain==1) {
32                         break;
33                     }
34
35                 }
36                 }while(!correct);
37         return guess;
38     }
39 }
40
```

Main.java | MainSover.java | MyGameNutpapat.java | NumberGame.java

```java
/* To change this license header, choose License Headers in Project Properties.
*/
package guessinggame;

import javax.swing.JOptionPane;
public class Main {
    public static void main(String[] args) {
        // upper limit for secret number in guessing game
        int upperBound = 100;
        int playAgain;
        do{
            NumberGame game = new MyGameNutpapat(upperBound);
            GameDialog gameDialog = new GameDialog();
            int solution = gameDialog.play( game );
            System.out.println("Answer is "+solution+" and count of your answer "+game.getCount());
            playAgain = JOptionPane.showConfirmDialog(null, "Want to play again", "Question", JOptionPane.YES_NO_OPTION);
        }while(playAgain == 0);


    }
}
```

MainSover.java | NumberGame.java

```java
package guessinggame;

public class MainSover {
    public static void main(String[] args) {
    // upper limit for secret number in guessing game
        int upperBound = 100;
        NumberGame game = new MyGameNutpapat(upperBound);
        GameConsole ui = new GameConsole();
        //show solver
        ui.play(game);

        System.out.println("Answer is "+((MyGameNutpapat)(game)).showSecret()+" and count of your answer "+game.getCount());
    }
}
```

NumberGame.java

```java
/**
 * The base class for all number games.
 * Your guessing game should extend this class and
 * override the methods: guess(), toString(), getUpperBound().
 *
 * Your class should not override getMessage() and setMessage(),
 * just use the methods from this class.
 */
package guessinggame;
public class NumberGame {
    /** A helpful message for user. */
    private String message;
        protected int count=0;

    /** Initialize a new default game. */
    public NumberGame() {
        // initialize your game.
        message = "";
    }

    /**
     * Evaluate a user's answer to the game.
     * @param answer is the user's answer, as an integer.
     * @return true if correct, false otherwise
     */
    public boolean guess(int answer) {
        // TODO your subclass should override this method
        message = "Sorry, that's not correct";
        return false;
    }

    /**
     * Return a message about the most recent call to guess().
     * Initially the message should tell the user something so
     * the user knows what to guess, such as
     * "I'm thinking of a number between 1 and xx".
     * @return string message related to the most recent guess.
     */
    public String getMessage() {
        return message;
    }

    /**
     * Set a message about the game.
     * @param newmessage a string about game or most recent guess.
     */
    public void setMessage(String newmessage) {
        this.message = newmessage;
    }

    /**
     * Get the largest possible value of the solution for this game.
     * For a guessing game, this should be the upper bound of secret.
     */
    public int getUpperBound() {
        return Integer.MAX_VALUE;  // not very helpful :-)
    }

    /**
     * toString describes the game or problem.
     * @return description of this game or the problem to be solved.
     */
    @Override
    public String toString() {
        return "You should override this method for your game";
    }
    public int getCount(){
        return 0;
    }
}
```