Implement a solution to the problem described below. Your work must include the following:

* Your version of the PlayGame.java, Player.java, Dealer.java
  + include as many comments as are necessary to describe what the given code segment is doing
  + make sure that your code is formatted properly
* Output of your program that shows multiple runs that test various scenarios
* Please make sure to include your name in each file that you are submitting

BEFORE STARTING PLEASE READ THE PROBLEM DESCRIPTION AND THE INSTRUCTIONS CAREFULLY.

**Problem Description**

Guessing game of OddOrEven is a game where a dealer uses a cup to roll two six-sided dice. The cup is placed upside down on a table so that the value of the dice is concealed. Players then have to guess if the sum of the dice values id ODD or EVEN.

We will develop a program that simulates a variation of this game. The simulated game will have a dealer and two players. The players will simply guess whether the sum of the dice values is odd or even. One point will be awarded to each player who correctly guesses the result. The game will play for five rounds, and the player with the most points is the grand winner. The tie is also possible.

In this program we will utilize the Die class that we studied before. We will create two objects of this class to represent two six-sided dice.

In addition, we will write the following classes:

* Dealer class: we will create an instance of this class to represent the dealer. It will have the ability to roll the dice, and report whether the total dice value is odd or even
* Player class: we will create two objects of this class to represent the players. Instances of the Player class can store the player's name, player's guess made between ODD or EVEN, and the running total of awarded points
* PlayGame class: its constructor creates all the necessary elements for the game, plays the round, and displays the running results and the final results
* PlayGameClient class: this class has main for this application and is fully implemented.

See the provided UML diagram, sample run and the javaDoc comments throughout the classes

**Sample Run of the program**

Enter the name of the first player

Jack

Enter the name of the second player

Jill

Playing round #1

The dealer rolled 3 2

Result ODD

Jack guessed EVEN

Jill guessed ODD

---> Awarding 1 point to Jill

Playing round #2

The dealer rolled 2 4

Result EVEN

Jack guessed ODD

Jill guessed ODD

Playing round #3

The dealer rolled 4 3

Result ODD

Jack guessed ODD

Jill guessed ODD

---> Awarding 1 point to Jack

---> Awarding 1 point to Jill

Playing round #4

The dealer rolled 5 2

Result ODD

Jack guessed ODD

Jill guessed EVEN

---> Awarding 1 point to Jack

Playing round #5

The dealer rolled 1 2

Result ODD

Jack guessed EVEN

Jill guessed EVEN

\*\*\* Game over. Here are the results \*\*\*

Jack scored 2 points.

Jill scored 2 points.

It is a tie!!!

Would you like to play again? (yes/no)

yes

Enter the name of the first player

Lady

Enter the name of the second player

Tramp

Playing round #1

The dealer rolled 5 4

Result ODD

Lady guessed ODD

Tramp guessed ODD

---> Awarding 1 point to Lady

---> Awarding 1 point to Tramp

Playing round #2

The dealer rolled 6 2

Result EVEN

Lady guessed EVEN

Tramp guessed EVEN

---> Awarding 1 point to Lady

---> Awarding 1 point to Tramp

Playing round #3

The dealer rolled 1 6

Result ODD

Lady guessed ODD

Tramp guessed EVEN

---> Awarding 1 point to Lady

Playing round #4

The dealer rolled 6 3

Result ODD

Lady guessed ODD

Tramp guessed EVEN

---> Awarding 1 point to Lady

Playing round #5

The dealer rolled 2 2

Result EVEN

Lady guessed ODD

Tramp guessed EVEN

---> Awarding 1 point to Tramp

\*\*\* Game over. Here are the results \*\*\*

Lady scored 4 points.

Tramp scored 3 points.

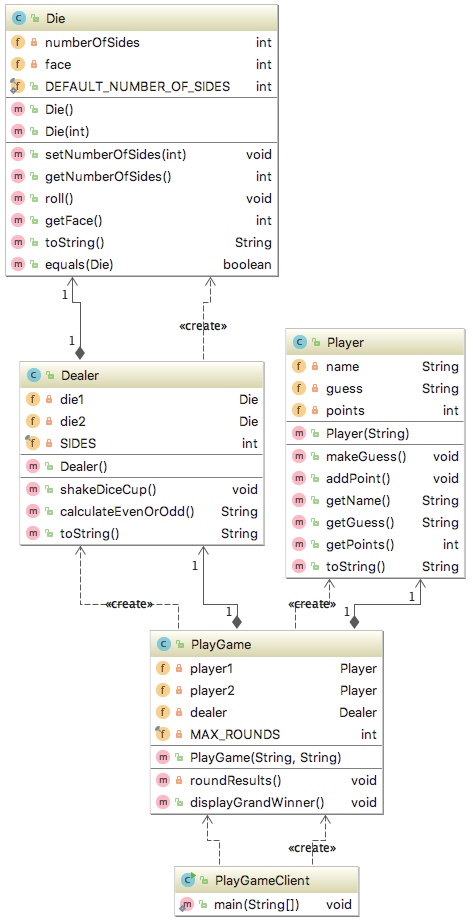
Lady is the winner!!!

Would you like to play again? (yes/no)

no

Goodbye!

**UML Diagram**

****