**BlackJack.java**

// NOTE ABOUT INSTRUCTIONS: variable names and method names referenced in the instructions  
// are placed in quotes for emphasis only. When using variable names and methods in your  
// code, you will not place them in quotes.  
//  
// Please remove all INSTRUCTION COMMENTS before submitting your  
// project to Moodle.  
//  
// ALL code that you are instructed to write MUST BE COMMENTED by you.  
// Commenting your code is a significant part of your grade on this assignment.  
//  
//\*\*\*  
//\*\*\* INSTRUCTIONS FOR CODE FOR YOU TO WRITE  
//\*\*\*  
//\*\*\*  
//\*\*\* 1) Define a public class named "BlackJack" that subclasses class "CardGame".  
//\*\*\* 2) Inside class "BlackJack", include the following:  
//\*\*\*  
//\*\*\* 1) Define an enumerated type by adding the following line of  
//\*\*\* code to the class:  
//\*\*\* enum Winner {DEALER, PLAYER, TIE}  
//\*\*\* NOTE: The semi-colon is not necessary at the end of this enum and  
//\*\*\* is considered redundant. You can use one if you like.  
//\*\*\* 2) Create one private property per the following:  
//\*\*\* a) Property is of datatype "Winner" and named "winner".  
//\*\*\* 3) Create a constant of datatype int, named "DEALER\_HIT\_STAY\_THRESHOLD",  
//\*\*\* having a value of 16.  
//\*\*\* 4) Create a constant of datatype String, named "DEALER\_NAME",  
//\*\*\* having a value of "Dealer".  
//\*\*\* 5) Create and assign values to two private reference variables  
//\*\*\* per the following:  
//\*\*\* a) First reference variable:  
//\*\*\* A) Datatype is "BlackJackPlayer" and named "player".  
//\*\*\* B) Using the no-arg constructor, create a new "BlackJackPlayer"  
//\*\*\* instance and assign to "player".  
//\*\*\* b) Second reference variable:  
//\*\*\* A) Datatype is "BlackJackPlayer" and named "dealer".  
//\*\*\* B) Using the one-arg constructor, create a new "BlackJackPlayer"  
//\*\*\* instance and assign to "dealer".  
//\*\*\* C) Pass the named constant "DEALER\_NAME" as the one argument  
//\*\*\* to the constructor.  
//\*\*\* 6) Create a public getter for each of these three properties "winner",  
//\*\*\* "player", and "dealer" that just returns the property.  
//\*\*\* 7) Create a private setter for property "winner" that assigns  
//\*\*\* the parameter variable to the property and returns no value.  
//\*\*\* (Standard setter method)  
//\*\*\* 8) Create a public method named "dealCardWithHit" that takes one  
//\*\*\* argument and returns no value.  
//\*\*\* a) The one argument is of datatype "Player" and named "p".  
//\*\*\* b) The method body consists of one line of code as follows:  
//\*\*\* A) Invoke the inherited method "dealCard" passing as the  
//\*\*\* argument to method "dealCard", the parameter variable "p".  
//\*\*\* 9) Implement the abstract method "determineOutcome" having  
//\*\*\* no method code. In other words, it is an empty method for now.  
//\*\*\* 10) Create a public one-arg constructor for class "BlackJack".  
//\*\*\* a) The one argument is of datatype int and named "numCards".  
//\*\*\* b) The method body consists of one line of code as follows:  
//\*\*\* A) Invoke the superclass' one-arg constructor passing  
//\*\*\* parameter variable "numCards" as the one argument.  
//\*\*\* 11) Create a public static inner class named "BlackJackPlayer" that  
//\*\*\* subclasses class "Player". Inner class means that it is nested  
//\*\*\* inside class "BlackJack".  
//\*\*\* 12) Inside class "BlackJackPlayer", include the following:  
//\*\*\*  
//\*\*\* 1) Create a constant of datatype int, named "BUST\_SCORE",  
//\*\*\* having a value of 21.  
//\*\*\* 2) Create two private properties per the following:  
//\*\*\* a) First property is of datatype int and named "numberHits".  
//\*\*\* b) Second property is of datatype boolean and named "bust".  
//\*\*\* 3) Create a public getter for each of these two properties that  
//\*\*\* just returns the property.  
//\*\*\* Remember: the getter for a boolean property uses "is" not "get".  
//\*\*\* 4) Create a private setter for each of these two properties that  
//\*\*\* assigns the parameter variable to the property and returns no value.  
//\*\*\* (Standard setter method)  
//\*\*\* 13) Create a public method named "checkBust" that takes no arguments  
//\*\*\* and returns boolean.  
//\*\*\* a) The method body consists of two lines of code as follows:  
//\*\*\* A) Declare a local variable of datatype boolean named "bust",  
//\*\*\* and assign the value of false to it.  
//\*\*\* B) Write a return statement that returns the variable "bust".  
//\*\*\* 14) Create a public no-arg constructor for class "BlackJackPlayer".  
//\*\*\* a) The method contains no code.  
//\*\*\* 15) Create a public one-arg constructor for class "BlackJackPlayer".  
//\*\*\* a) The one argument is of datatype String and named "name".  
//\*\*\* b) Invoke the superclass' one-arg constructor passing  
//\*\*\* parameter variable "name" as the one argument.  
//\*\*\*