Hi there! I have a question about updating the blackjack.py file. I see in the instructions we should add two new methods. Would it be okay to modify other methods as well? Specifically, I was having a hard time getting the bet amount in the GUI to update properly with my former getBet(self) in blackjack.py. I changed that method to getBet(self, bet) and everything seems to be working

-

It should be ok, but there is no need.

The change in Phase 3 should use the GUI to ask user for a valid bet amount,

and then can simply set the bet on the Blackjack object: self.game.bet = validBet

Of course, three buttons – Play, Hit and Stand - do nothing.

You will implement helper methods and three event handlers for these buttons:

**displayPlayer(self),**

**displayDealer(self):**

Helper methods that can be used to update the various text entry fields.

Use these in the event handlers you will be writing.

**displayResult(self):**

Helper method to update the result of the game.

It calls the determineOutcome method to decide what the outcome of the game is.

It also updates the result and the money fields in UI. Useful in the event handlers.

**playerCanPlayTurn(self):**

Helper method to check if the game is underway so that player can play a turn.

Useful in the event handlers.

**play(self):**

Method to start a new game.

Method checks that the game is not already underway,

if so, gives feedback in the results field and returns.

Otherwise, verifies that the bet amount is valid.

If not valid, gives feedback in the results field and returns. self.game.bet = validBet

If a game is not underway and the bet amount is valid,

starts the game by setting the bet on the game object,

calling setupRound and

displaying player and dealer’s cards and points.

Finally, before returning, the method checks for a blackjack and takes needed actions.

**hit(self):**

Method confirms that user can play a turn.

If not returns

else calls the takePlayerTurn

and reports the player state by calling displayPlayer.

Next checks if the player is busted,

if so, ends the game

and updates the result by calling displayResult.

**stand(self):**

Method confirms that user can play a turn,

if not returns.

Otherwise ends the game

and has the dealer play his turn.

It then reports the dealer state by calling displayDealer.

Finally updates the result by calling displayResult.