

Class Name:	GameManager		
Method Name (Test Case)	Purpose	Result	Correction/Action/Notes
SwitchToNextPlayer	Ensure that the method moves to the next player when the player is not the last player.	Failed	I had to set the Players property in PlayerManager as virtual.
SwitchToNextPlayer	-	Failed	I forgot to set the first player IsFinished value to true, so the method did not move to the next player.
SwitchToNextPlayer	-	Passed	Results as expected.
CheckIfRoundOver	Ensure that the method returns true if all players and the dealer are finished.	Failed	I had to set the Dealer property in PlayerManager as virtual.
CheckIfRoundOver	-	Failed	The CheckIfRoundOver checks the dealers IsFinished value through a private instance variable which can't be updated through the test method. I changed the code in CheckIfRoundOver so that it checks playerManager.Dealer instead, which is reachable through the test method.
CheckIfRoundOver	-	Passed.	Results as expected.

Class Name:	PlayerManager		
Method Name (Test Case)	Purpose	Result	Correction/Action/Notes
GetCurrentPlayer	Ensure that the method returns the first player whose IsFinished value is false.	Failed	The PlayerManager class did not initialize the "Players" property (only the "players" field which is not reachable from the test method) at initialization. I had to change this.
GetCurrentPlayer	-	Failed	I forgot to update the GetCurrentPlayer method so that it uses the property instead of the field.
GetCurrentPlayer	-	Passed	Results as expected.
CheckIndex	Ensure that the method returns false if the index is higher than the player count.	Failed	The PlayerCount property handles the "player" field and not the "Player" property.
CheckIndex	-	Passed	Results as expected.

Class Name:	PlayerRepository		
Method Name (Test Case)	Purpose	Result	Correction/Action/Notes
AddPlayer	Ensure that the method adds a player to the database.	Passed	Results as expected.
DeletePlayer	Ensure that the method deletes a player from the database.	Passed	Results as expected.

Class Name:	DeckRepository		
Method Name (Test Case)	Purpose	Result	Correction/Action/Notes
AddDeck	Ensure that the method adds a deck to the database.	Failed	Did not initialize the testDeck properly before running my test.
AddDeck	-	Failed	Could not remove the Deck in my TearDown method since it's tied to Cards in the database.
AddDeck	-	Failed	Tried to remove associated cards in the TearDown method but the test still fails. I had to load the associated cards properly before attempting to delete them.
AddDeck	-	Passed	Results as expected.
GameDeck	Ensure that the method returns a deck with all belonging values and properties.	Passed	Results as expected.