Checkers Game Data Model

By Benjavicha Hotrabhavananda

Piece class:

color: String

Methods:

display()

tracePaths(sourceX, sourceY, destinationX, destinationY): PathTrace

---

To hold all the pieces history

King class extends Piece:

kingPiece: Piece

Methods:

tracePaths(sourceX, sourceY, destinationX, destinationY): PathTrace

---

This King Piece extends out from Piece class with backward ability.

CheckerGame Class:

Board : CheckerBoard

Player1 : Player

Player2 : Player

moveHistory : Map<Player, List<int[2][2]>>

selectedPieceColor : Map<Player,String>

Methods:

selectPieceType(Player) : void

startTheGame() : void

announceWinner() : Player

In the CheckerGame, this class determine if a move is illegal and a winner, loser, or a draw condition.

Player Class:

playerName : String

numOfWins : int

numOfLoss : int

numOfDraws : int

matchesPlayed : List<String>

Methods:

getPlayerName()

setPlayerName()

getNumOfWins()

setNumOfWins()

getNumOfLoss()

setNumOfLoss()

getNumOfDraws()

setNumOfDraws()

This class to keep track of play information.

Board class:

rectangle : Box[8][8]

removedPieces : List<Piece>

Methods:

movePiece(pieceColor, sourceX, sourceY, destinationX, destination) : boolean

checkForCheck() : boolean

removePiece(sourceX, sourceY)

This class determines what moves are possible/allowable.
