Package Model

Chessboard

#Piece[][] board
+King whiteKing
+King blackKing
+Class<?> promotionType

hasPiece(int row, int col): boolean
hasPiece(String t, int row, int col): boolean
getPiece(int row, int col): Piece
+move(int r1, c1, int r2, c2)
+printboard()
checkCheck(): King
checkCheckMate(King king): boolean
-Promote(int row, int col)

Has

2-32

Has

Piece (abstract)

+String color
+String type
+int row, col
+boolean moved

getColor(): String
getType(): String
move(int row, int col)
+Piece()
+Piece(String color, int row, int col)
+upperRightCheck(int row, int col):boolean
+upperLeftCheck(int row, int col):boolean
+lowerLeftCheck(int row, int col):boolean
+lowerRightCheck(int row, int col):boolean
+upperVertCheck(int row, int col):boolean
+lowerVertCheck(int row, int col):boolean
+leftHorizontalCheck(int row, int col):boolean
+rightHorizontalCheck(int row int col):boolean

Pawn
Knight
Bishop
Rook
Queen

King

-boolean inCheck

+checkStatus(): boolean
+validMoveCount():int
+checkThreats(int r, int c):boolean
+checkThreats():boolean
+move(int row, int col): boolean

IllegalMoveException

+IllegalMoveException(String s)

Package Game

Chess

+boolean whiteTurn
+boolean blackTurn
+Chessboard board
+boolean draw

+validinput(String s): boolean
+textToCoord(String s): int[]
-convertToPieceType(String s):Class<?>
-canSaveKing(King k, Piece p):boolean
-checkCheckMate(King def):boolean