Class ChessBoard

* ChessPiece[][] board

————————————

* <<constructor>>ChessBoard()
* initialize()
* getPiece(String position): ChessPiece {exception IllegalPositionException}
* placePiece(ChessPiece piece, String position): boolean
* move(String fromPosition, String toPosition) {IllegalMoveException}
* toString()

Class *ChessPiece* {abstract}

* enum Color {White,Black}
* ChessBoard board
* int row
* int column
* Color color
* String position

————————————

* <<constructor>> ChessPiece(ChessBoard board, Color color)
* getColor(): Color
* getPosition(): String
* setPosition()
* *legalMoves*(): ArrayList<String>
* *toString*(): String

Class King

* ArrayList<String> legalMoves

———————————

* <<constructor>> King(ChessBoard board, Color color)
* legalMoves(): ArrayList<String>
* toString(): String

Class Pawn

* ArrayList<String> legalMoves

———————————

* <<constructor>> Pawn(ChessBoard board, Color color)
* legalMoves(): ArrayList<String>
* toString(): String

Class Bishop

* ArrayList<String> legalMoves

———————————

* <<constructor>> Bishop(ChessBoard board, Color color)
* legalMoves(): ArrayList<String>
* toString(): String

Class Rook

* ArrayList<String> legalMoves

———————————

* <<constructor>> Rook(ChessBoard board, Color color)
* legalMoves(): ArrayList<String>
* toString(): String