XXLChess

App

+ final SPRITESIZE: int
+ final CELLSIZE: int
+ final SIDEBAR: int
+ final BOARD_WIDTH: int
+ WIDTH: int
+ HEIGHT: int
+ final FPS: int
+ configPath: String
+ pieces: HashMap<Character, PImage>
+ pieceCode: HashMap<Character, Piece>
+ pieceMoves: HashMap<Character, int[][]>
+ board: char[][]
+ player1Time: int
+ player2Time: int
+ player1Increment: int
+ player2Increment: int
+ frameCount: int
+ playerColour: String
+ WhitePawn: WhitePawn
+ BlackPawn: BlackPawn
+ Camel: Camel
+ Chancellor: Chancellor
+ Amazon: Amazon
+ Archbishop: Archbishop
+ Rook: Rook
+ Bishop: Bishop
+ Queen: Queen
+ Guard: Guard
+ Knight: Knight
+ King: King
+ rowChosen: int
+ colChosen: int
+ lastRow: int
+ lastCol: int
+ lastPieceRow: int
+ lastPieceCol: int
+ pieceSelected: boolean
+ moved: boolean
+ player1Turn: boolean
+ ended: boolean
+ wKingMoved: boolean
+ wLRookMoved: boolean
+ wRRookMoved: boolean
+ bKingMoved: boolean
+ bLRookMoved: boolean
+ bRRookMoved: boolean
+ autoOn: boolean
+ flash: boolean
+ flashCount: int
+ path: int[][]
+ movables: ArrayList<Character>
+ pick: Random

+ settings(): void
+ setup(): void
+ keyPressed(): void
+ mousePressed(): void
+ draw(): void
+ timeString(time: int): String
+ movePiece(startX: int, startY: int, endX: int, endY: int): void
+ drawPath(): void
+ drawBoard(): void
+ drawPieces(): void
+ getValidMove(rowChosen: int, colChosen: int, board: char[][]): int[][]
+ drawValidMove(): void
+ isEnemy(piece1: char, piece2: char): boolean
+ check(king: char, board: char[][]): boolean
+ drawCheck(king: char): void
+ vmCheckLimit(rowChosen: int, colChosen: int): int[][]
+ checkmate(king: char): boolean
+ autoScan(board: char[][]): char[]
+ autoMove(): void
+ testing(): void
+ static main(args: String[]): void

Use

King
+ static final wImage: String
+ static final bImage: String
+ validMove(row: int, col: int): int[][]

Amazon
+ static final wImage: String
+ static final bImage: String
+ validMove(row: int, col: int): int[][]

Archbishop
+ static final wImage: String
+ static final bImage: String
+ validMove(row: int, col: int): int[][]

Guard
+ static final wImage: String
+ static final bImage: String
+ validMove(row: int, col: int): int[][]

Camel
+ static final wImage: String
+ static final bImage: String
+ validMove(row: int, col: int): int[][]

WhitePawn
+ static final wImage: String
+ static final bImage: String
+ validMove(row: int, col: int): int[][]

Piece
+ validMove(row: int, col: int): int[][]

Knight
+ static final wImage: String
+ static final bImage: String
+ validMove(row: int, col: int): int[][]

Rook
+ static final wImage: String
+ static final bImage: String
+ validMove(row: int, col: int): int[][]

Bishop
+ static final wImage: String
+ static final bImage: String
+ validMove(row: int, col: int): int[][]

Queen
+ static final wImage: String
+ static final bImage: String
+ validMove(row: int, col: int): int[][]

Chancellor
+ static final wImage: String
+ static final bImage: String
+ validMove(row: int, col: int): int[][]

BlackPawn
+ static final wImage: String
+ static final bImage: String
+ validMove(row: int, col: int): int[][]