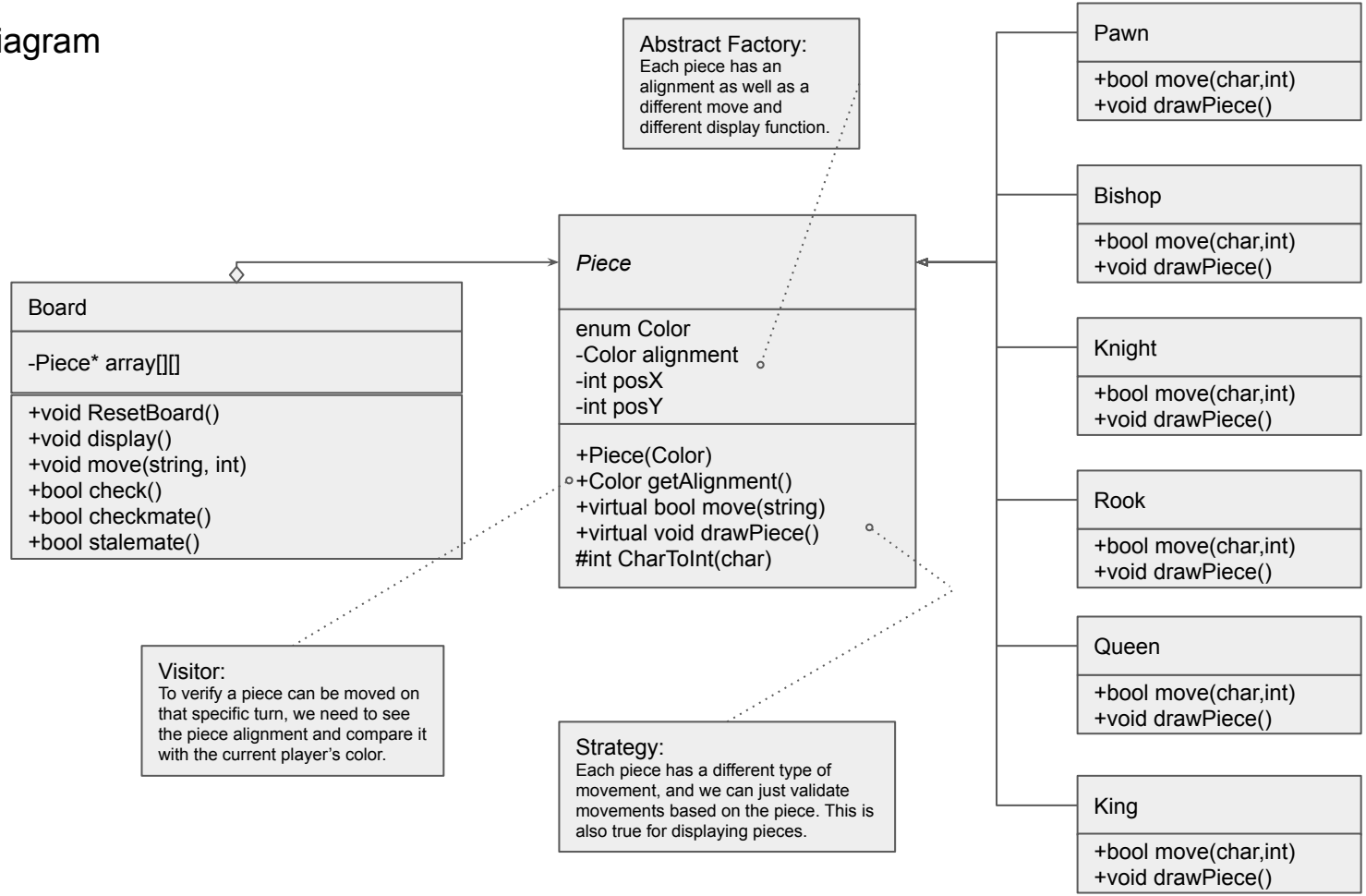
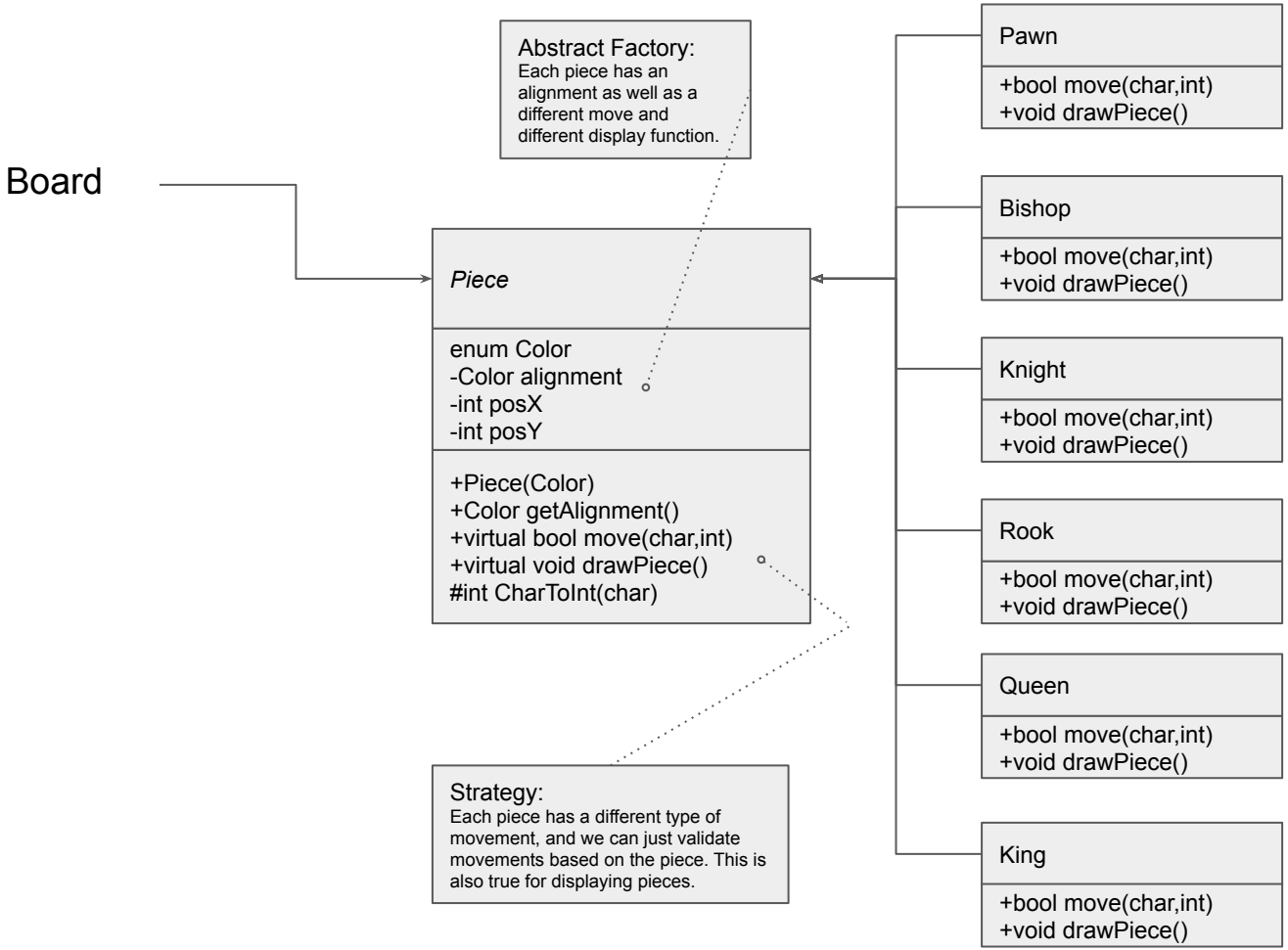
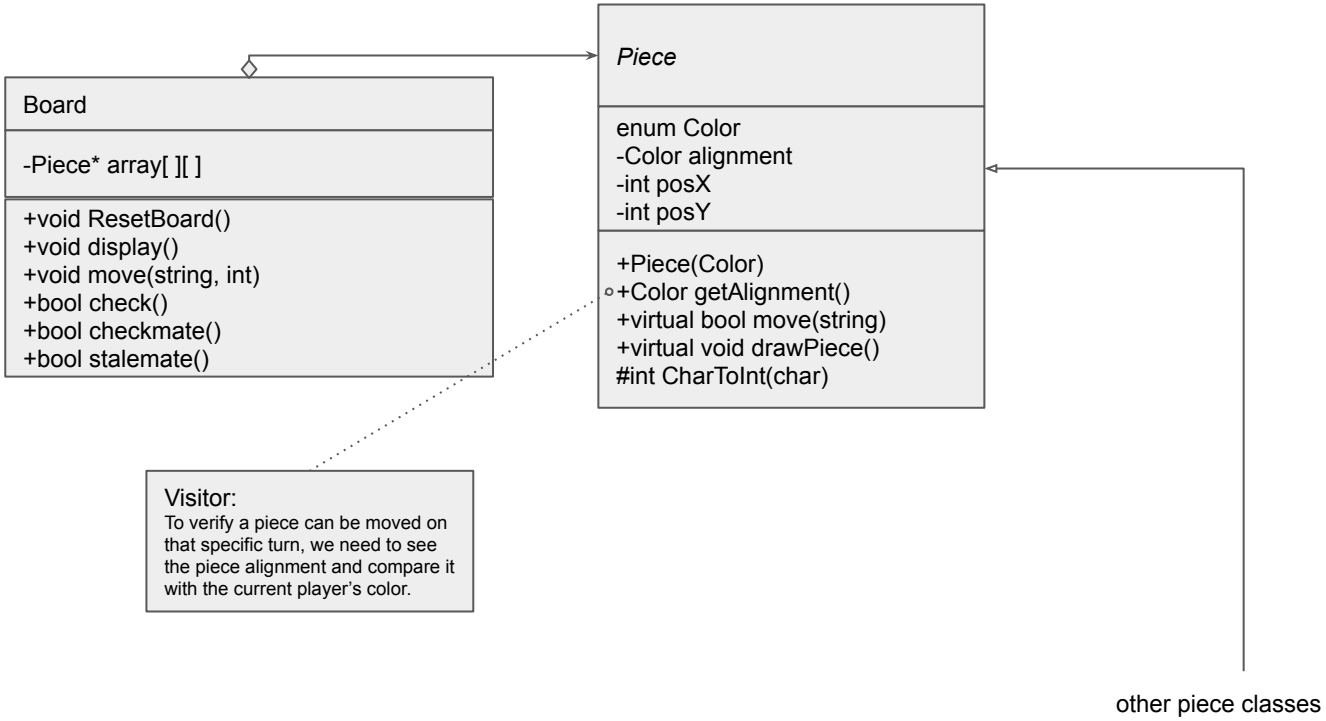
Class Diagram




Board
Abstract Factory:
Each piece has an alignment as well as a different move and different display function.
Piece
enum Color
-Color alignment
-int posX
-int posY
+Piece(Color)
+Color getAlignment()
+virtual bool move(char,int)
+virtual void drawPiece()
#int CharToInt(char)
Strategy:
Each piece has a different type of movement, and we can just validate movements based on the piece. This is also true for displaying pieces.
Pawn
+bool move(char,int)
+void drawPiece()
Bishop
+bool move(char,int)
+void drawPiece()
Knight
+bool move(char,int)
+void drawPiece()
Rook
+bool move(char,int)
+void drawPiece()
Queen
+bool move(char,int)
+void drawPiece()
King
+bool move(char,int)
+void drawPiece()


Board
-Piece* array[ ][ ]
+void ResetBoard()
+void display()
+void move(string, int)
+bool check()
+bool checkmate()
+bool stalemate()
Piece
enum Color
-Color alignment
-int posX
-int posY
+Piece(Color)
+Color getAlignment()
+virtual bool move(string)
+virtual void drawPiece()
#int CharToInt(char)
Visitor:
To verify a piece can be moved on that specific turn, we need to see the piece alignment and compare it with the current player's color.
other piece classes