```java
package Project3;

/**********************************************************************
 * CIS 163 Section 01
 * Project 3: Chess Game
 * Pawn Class
 *
 * This class represents a ChessPiece that is a pawn.
 *
 * @author George Fayette
 * @version 3/23/2019
 *********************************************************************/
public class Pawn extends ChessPiece {

    /**
     * Public boolean representing whether or not the pawn has moved.
     */
    public boolean firstMove;

    /******************************************************************
     * Public constructor sets player to parameter value.
     * @param player The player type.
     *****************************************************************/
    public Pawn(Player player) {
        super(player);
        firstMove = true;
    }

    /******************************************************************
     * Public String, returns the ChessPiece type.
     * @return A string representing the ChessPiece type.
     *****************************************************************/
    public String type() {
        return "Pawn";
    }

    /******************************************************************
     * Public boolean, returns true if the move is valid.
     * @param move The move that is being checked.
     * @param board The array of IChessPieces that is being checked.
     * @return True if the move is valid.
     *****************************************************************/
    public boolean isValidMove(Move move, IChessPiece[][] board) {
        boolean valid = true;
        if (!super.isValidMove(move, board)) {
            valid = false;
        } else {

            IChessPiece moveFrom = board[move.fromRow][move.fromColumn];
            IChessPiece moveTo = board[move.toRow][move.toColumn];
            int vDistance = move.toRow - move.fromRow;
            int hDistance = move.toColumn - move.fromColumn;
            int maxDistance = 1;

            if (firstMove) {
                maxDistance = 2;
            }

            if (moveFrom.player() == Player.WHITE) {
                if (hDistance == 0) {
                    if (vDistance < -maxDistance || vDistance > 0) {
                        valid = false;
                    }
                    for (int i = move.fromRow - 1; i >= move.toRow;
                         --i) {
                        if (board[i][move.fromColumn] != null) {
                            valid = false;
                        }
                    }
                } else if ((hDistance != 1 && hDistance != -1) ||
                        vDistance != -1 || moveTo == null) {
                    valid = false;

                }
            } else if (moveFrom.player() == Player.BLACK) {
                if (hDistance == 0) {
                    if (vDistance > maxDistance || vDistance < 0) {
                        valid = false;
                    }
                    for (int i = move.fromRow + 1; i <= move.toRow;
                         ++i) {
                        if (board[i][move.fromColumn] != null) {
                            valid = false;
                        }
                    }
                } else if ((hDistance != 1 && hDistance != -1) ||
                        vDistance != 1 || moveTo == null) {
                    valid = false;

                }
            }
        }
        return valid;
    }
}
```