```java
import java.util.Scanner;

public class TicTacToe {

    int[][] board = new int[3][3]; // main tictactoe board
    final int BLANK = 0;
    final int X_MOVE = 1;
    final int O_MOVE = 2;
    final int X_TURN = 0;
    final int O_TURN = 1;
    int turn = X_TURN;
    Scanner scanner;
    String input = "";
    int X_WINS = 0;
    int O_WINS = 0;

    public TicTacToe() {
        scanner = new Scanner(System.in);
        boolean stillPlaying = true;
        while (stillPlaying == true) {
            while (checkWin(X_MOVE) == false && checkWin(O_MOVE) == false &&
checkTie() == false) { // this ends the while loop if it finds that someone has won
or it is a tie
                printBoard();
                input = scanner.nextLine();
                if (input.length() != 2) { // if and else if statements guarantee
no errors occur when using the scanner along with guiding the player to tell them
what they should input.
                    System.out.println("Enter a number followed by a letter");
                } else if (input.charAt(0) != 'a' && input.charAt(0) != 'b' &&
input.charAt(0) != 'c') {
                    System.out.println("Row must be an a, b, or c.");
                } else if (input.charAt(1) != '1' && input.charAt(1) != '2' &&
input.charAt(1) != '3') {
                    System.out.println("Column must be a 1, 2, or 3.");
                } else {
                    int row = input.charAt(0) - 'a';
                    int column = input.charAt(1) - '1';
                    if (board[row][column] == BLANK) {
                        if (turn == X_TURN) {
                            board[row][column] = X_MOVE;
                            turn = O_TURN;
                        } else {
                            board[row][column] = O_MOVE;
                            turn = X_TURN;
                        }
                    } else {
                        System.out.println("There is a piece there!"); // makes
sure you cant play in a taken area
                    }
                }
            }
            printBoard();
            if (checkWin(X_MOVE) == true) {
                System.out.println("X wins!");
                X_WINS++;
            } else if (checkWin(O_MOVE) == true) {
                System.out.println("O wins!");
                O_WINS++;
            }
            else {
                System.out.println("It's a tie!");
            }
            System.out.println("X has won " + X_WINS + " times and O has won "
+O_WINS + " times."); // display amount of wins both players have
            System.out.println("Play again? ");
            String yesno = scanner.nextLine(); // play again scanner
            if (yesno.equals("yes") || yesno.equals("y")) { // resetting the board
if they want to play again
                stillPlaying = true;
                board[0][0] = BLANK;
                board[0][1] = BLANK;
                board[0][2] = BLANK;
                board[1][0] = BLANK;
                board[1][1] = BLANK;
                board[1][2] = BLANK;
                board[2][0] = BLANK;
                board[2][1] = BLANK;
                board[2][2] = BLANK;
            }
            else {
                stillPlaying = false; // ends the while loop, thus, ending the code
            }
        }
    }
    public void printBoard() { // printing the board
        System.out.println("\t1\t2\t3");
        for (int row = 0; row < board.length; row++) {
            String output = (char) ('a' + row) + "\t";
            for (int column = 0; column < board[0].length; column++) {
                if (board[row][column] == BLANK) {
                    output += "\t";
                } else if (board[row][column] == X_MOVE) {
                    output += "X\t";
                } else if (board[row][column] == O_MOVE) {
                    output += "O\t";
                }
            }
            System.out.println(output);
        }
    }
    public boolean checkWin(int player) { // checking the wins
        if (board[0][0] == player && board[0][1] == player && board[0][2] ==
player) {
            return true;
        }
        if (board[1][0] == player && board[1][1] == player && board[1][2] ==
player) {
            return true;
        }
        if (board[2][0] == player && board[2][1] == player && board[2][2] ==
player) {
            return true;
        }
        if (board[0][0] == player && board[1][0] == player && board[2][0] ==
player) {
            return true;
        }
        if (board[0][1] == player && board[1][1] == player && board[2][1] ==
player) {
            return true;
        }
        if (board[0][2] == player && board[1][2] == player && board[2][2] ==
player) {
            return true;
        }
        if (board[0][0] == player && board[1][1] == player && board[2][2] ==
player) {
            return true;
        }
        if (board[2][0] == player && board[1][1] == player && board[0][2] ==
player) {
            return true;
        }
        return false;
    }
    public boolean checkTie() { // checking the tie by checking if there are any
blank pieces on the board. if there no blank pieces, it returns true which causes a
tie as seen from the while loop in TicTacToe() method
        for (int row = 0; row < board.length; row++) {
            for (int column = 0; column < board[0].length; column++) {
                if (board[row][column] == BLANK) {
                    return false;
                }
            }
        }
        return true;
    }

    public static void main(String[] args) {

        new TicTacToe();
    }

}
```