import java.util.Random;

import java.util.Scanner;

public class TTTB {

public static final char getComputerMove = 'O';

public static void main(String[] args) {

Scanner in = new Scanner(System.in);

char [][] board = new char[3][3];

boolean var = false;

int i = 0;

//Initializes board with dashes

while(i == 0) {

initializeBoard(board);

do {

printBoard(board);

System.out.println("\n\n");

getplayerMove(board);

printBoard(board);

System.out.println("\n\n");

var = win(board, 'X');

if(tie(board)) {

System.out.println("There was a tie");

break;

}

if(var) {

System.out.println("Player one won!!!!");

break;

}

getComputerMove(board);

System.out.println("\n\n");

var = win(board, 'O');

if(var) {

System.out.println("Computer won you suck!!!!");

break;

}

}while(!var);

System.out.println("Would you like to play again (0 for yes, 1 for no)?");

i =in.nextInt();

}

}

public static void printBoard(char[][] board) {

for(int i = 0; i < 3; i++) {

for(int j = 0; j < 3; j++) {

System.out.print(board[i][j]);

if(j != 2) {

System.out.print(" | ");

}

}

if(i !=2) {

System.out.println("\n--------------");

}

}

}

public static void initializeBoard(char [][]board) {

for(int i = 0; i < 3; i++) {

for(int j = 0; j < 3; j++) {

board[i][j] = '-';

}

}

}

public static boolean win(char[][] board, char player ) {

boolean H = winnerH(board, player);

boolean D = winnerD(board, player);

boolean V = winnerV(board, player);

if(H == true || D == true || V == true) {

return true;

}

else

return false;

}

public static boolean winnerH(char [][]board, char player) {

if((board [0][0] == player && board [0][0] == board [0][1] && board [0][1] == board [0][2]) ||

(board [1][0] == player && board [1][0] == board [1][1] && board [1][1] == board [1][2]) ||

(board [2][0] == player && board [2][0] == board [2][1] && board [2][1] == board [2][2]))

return true;

else

return false;

}

public static boolean winnerD(char [][]board, char player) {

if((board [0][0] == player && board [0][0] == board [1][1] && board [1][1] == board [2][2]) ||

(board [2][0] == player && board [2][0] == board [1][1] && board [1][1] == board [0][2]))

return true;

else

return false;

}

public static boolean winnerV(char [][]board, char player) {

if((board [0][0] == player && board [0][0] == board [1][0] && board [1][0] == board [2][0]) ||

(board [0][1] == player && board [0][1] == board [1][1] && board [1][1] == board [2][1]) ||

(board [0][2] == player && board [0][2] == board [1][2] && board [1][2] == board [2][2]))

return true;

else

return false;

}

public static void getplayerMove(char [][]board) {

Scanner in = new Scanner(System.in);

int row = 0;

int col = 0;

boolean rt = true;

do {

System.out.println("\nEnter your row: ");

row = in.nextInt();

row = row -1;

System.out.println("Enter your col: ");

col = in.nextInt();

col = col -1;

if(board[row][col] == '-') {

board[row][col] = 'X';

rt = false;

}

else {

rt = true;

}

}while(rt == true);

}

public static void getComputerMove(char [][]board) {

Random rand = new Random();

int i = 0;

int j = 0;

boolean finish = false;

do{

i = rand.nextInt(3);

j = rand.nextInt(3);

if(board[i][j] == '-')

{

board[i][j] = 'O';

finish = true;

}

}while(finish != true);

}

public static boolean tie(char[][]board) {

boolean var = true;

for(int i = 0; i < 3; i++) {

for(int j = 0; j < 3; j++) {

if(board[i][j] == '-') {

var = false;

}

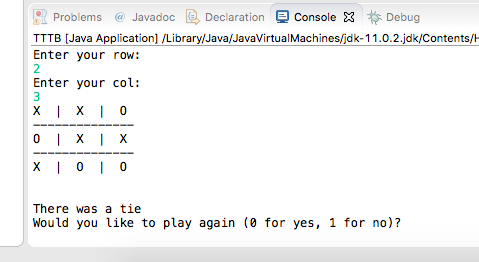
}

}

return var;

}

}



For example using the correct operator matters prefix or post-fix.

Int a = 10;

System.out.println(a++); prints 10

System.out.println(++a); prints 12

This occurs because a is first it prints 10 then adds 1, following that it adds 1 to a then prints a again.

A non-static method belongs to an object of the class, while static belongs to the class. Static methods cannot call static methods because it uses an instance variable state. Which causes an error in the compiler.