

Name= Shahid abbas

Semester=2nd spring BSCS

ID =16694

Subject = OOPs

Instructor= Muhammad Ayub khan

Semester Assignment SP20

Qno.1 Create Tic Tac Toe game in java (use any java tool for coding) and explain it in detail including screenshots ?

Ans:

```
import java.util.Arrays;
```

```
import java.util.InputMismatchException;
```

```
import java.util.Scanner;
```

```
/**
```

```
*
```

```
* @author shahid abbas
```

```
*
```

```
*/
```

```
public class TicTacToe {
```

```
    static Scanner in;
```

```
    static String[] board;
```

```
    static String turn;
```

```
    public static void main(String[] args) {
```

```
        in = new Scanner(System.in);
```

```
board = new String[9];
turn = "S";
String winner = null;
populateEmptyBoard();

System.out.println("Welcome Players to Tic Tac Toe.");
System.out.println("-----");
printBoard();
System.out.println("S's will play first. Enter a slot number to place S in:");

while (winner == null) {
    int numInput;
    try {
        numInput = in.nextInt();
        if (!(numInput > 0 && numInput <= 9)) {
            System.out.println("Invalid input; re-enter slot number:");
            continue;
        }
    } catch (InputMismatchException e) {
        System.out.println("Invalid input; re-enter slot number:");
        continue;
    }
    if (board[numInput - 1].equals(String.valueOf(numInput))) {
        board[numInput - 1] = turn;
        if (turn.equals("S")) {
            turn = "H";
        } else {
            turn = "S";
        }
    }
}
```

```
    }  
    printBoard();  
    winner = checkWinner();  
} else {  
    System.out.println("Slot already taken; re-enter slot number:");  
    continue;  
}  
}  
if (winner.equalsIgnoreCase("draw")) {  
    System.out.println("It's a draw! Thanks for playing.");  
} else {  
    System.out.println("Congratulations! " + winner + "'s have won! Thanks for playing.");  
}  
}
```

```
static String checkWinner() {  
    for (int a = 0; a < 8; a++) {  
        String line = null;  
        switch (a) {  
            case 0:  
                line = board[0] + board[1] + board[2];  
                break;  
            case 1:  
                line = board[3] + board[4] + board[5];  
                break;  
            case 2:  
                line = board[6] + board[7] + board[8];  
                break;
```

```
    case 3:
        line = board[0] + board[3] + board[6];
        break;
    case 4:
        line = board[1] + board[4] + board[7];
        break;
    case 5:
        line = board[2] + board[5] + board[8];
        break;
    case 6:
        line = board[0] + board[4] + board[8];
        break;
    case 7:
        line = board[2] + board[4] + board[6];
        break;
}
if (line.equals("SSS")) {
    return "S";
} else if (line.equals("HHH")) {
    return "H";
}
}

for (int a = 0; a < 9; a++) {
    if (Arrays.asList(board).contains(String.valueOf(a + 1))) {
        break;
    } else if (a == 8) {
        return "draw";
    }
}
```

```

    }
}

System.out.println(turn + "'s turn; enter a slot number to place " + turn + " in:");
return null;
}

static void printBoard() {
    System.out.println("");
    System.out.println("| " + board[0] + " | " + board[1] + " | " + board[2] + " |");
    System.out.println("|-----|");
    System.out.println("| " + board[3] + " | " + board[4] + " | " + board[5] + " |");
    System.out.println("|-----|");
    System.out.println("| " + board[6] + " | " + board[7] + " | " + board[8] + " |");
    System.out.println("");
}

static void populateEmptyBoard() {
    for (int a = 0; a < 9; a++) {
        board[a] = String.valueOf(a + 1);
    }
}
}

```

OUT PUT:

Output - Tic_Tac_Toe_Game (run)

```
ant -f C:\\Users\\wasimAbbas\\Documents\\NetBeansProjects\\Tic_Tac_Toe_Game -Dnb.internal.action.name=run run
init:
Deleting: C:\\Users\\wasimAbbas\\Documents\\NetBeansProjects\\Tic_Tac_Toe_Game\\build\\built-jar.properties
deps-jar:
Updating property file: C:\\Users\\wasimAbbas\\Documents\\NetBeansProjects\\Tic_Tac_Toe_Game\\build\\built-jar.properties
Compiling 1 source file to C:\\Users\\wasimAbbas\\Documents\\NetBeansProjects\\Tic_Tac_Toe_Game\\build\\classes
compile:
run:
Welcome Players to Tic Tac Toe.
-----

| 1 | 2 | 3 |
|-----|
| 4 | 5 | 6 |
|-----|
| 7 | 8 | 9 |

S's will play first. Enter a slot number to place S in:
4

| 1 | 2 | 3 |
|-----|
| S | 5 | 6 |
|-----|
| 7 | 8 | 9 |

H's turn; enter a slot number to place H in:
3

| 1 | 2 | H |
|-----|
| S | 5 | 6 |
|-----|
| 7 | 8 | 9 |
```

```
S's turn; enter a slot number to place S in:
```

```
5
```

```
| 1 | 2 | H |
|-----|
| S | S | 6 |
|-----|
| 7 | 8 | 9 |
```

```
H's turn; enter a slot number to place H in:
```

```
7
```

```
| 1 | 2 | H |
|-----|
| S | S | 6 |
|-----|
| H | 8 | 9 |
```

```
S's turn; enter a slot number to place S in:
```

```
6
```

```
| 1 | 2 | H |
|-----|
| S | S | S |
|-----|
| H | 8 | 9 |
```

```
Congratulations! S's have won! Thanks for playing.
```

```
BUILD SUCCESSFUL (total time: 39 seconds)
```

Explanation:

In this game I have take static Scanner in, static String[] board, static String turn. Scanner is predefine class and I have make it objects as in, String[] board is an array of length 9 elements and turn is string type variable which will used for player one and two turn. Winner is a variable of player who will win the game and a congratulation message will display for winner player.

numInput = in.nextInt(); this statement is used to take next input for player turn. Then I have used if for the decision for input number and player turn.

static String checkWinner() {} is a string type function in which I have used switch for making decision. I have make cases for different decision. The cases is used for each line input as vertically and horizontally.

```
if (line.equals("SSS")) {  
    return "S";  
} else if (line.equals("HHH")) {  
    return "H";  
}
```

This block of code is used for making decision for three time SSS or HHH in a line if SSS or HHH occur horizontally ,vertically or diagonally then it will show result for winner. printBoard() this function is used for design of the game. And populateEmptyBoard() function is used to populate board[] array.