```java
import java.awt.font.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class TicTacToe {

        private static int[][] winCombinations = new int[][] {
                        {0, 1, 2}, {3, 4, 5}, {6, 7, 8}, //horizontal wins
                        {0, 3, 6}, {1, 4, 7}, {2, 5, 8}, //vertical wins
                        {0, 4, 8}, {2, 4, 6}                     //diagonal wins
        };

        private static JButton buttons[] = new JButton[9]; //create 9 buttons



        public static void main (String[] args)
        {
                gamePanel(); //launch game
        }

        private static void gamePanel(){
                JFrame frame = new JFrame ("Tic Tac Toe");
                frame.setDefaultCloseOperation (JFrame.EXIT_ON_CLOSE);


                JPanel panel = new JPanel(); //creating a panel with a box like a tic tac toe board
                panel.setLayout (new GridLayout (3, 3));
                panel.setBorder (BorderFactory.createLineBorder (Color.gray, 3));
                panel.setBackground (Color.white);

                for(int i=0; i<=8; i++){ //placing the button onto the board
                        buttons[i] = new MyButton();
                        panel.add(buttons[i]);
                }

                frame.getContentPane().add (panel);
                frame.pack();
                frame.setVisible(true);
                frame.setSize(500, 500);// set frame size and let teh game begin
        }

        public static int xOrO=0; // used for counting

        private static class MyButton extends JButton
        implements ActionListener {//creating own button class because JButton sucks:)

                int again=1000;//set again at 1000 so we don't make the mistake we can play again
                boolean win=false; // there is not a win
                String letter; // x or o
                public MyButton() {     // creating blank board
                        super();
                        letter=" ";
                        setFont(new Font("Dialog", 1, 60));
                        setText(letter);
                        addActionListener(this);
                }
                public void actionPerformed(ActionEvent e) { // placing x or o's
                        if((xOrO%2)==0 && getText().equals(" ") && win==false){
                                letter="X";
                                xOrO=xOrO+1;
                                System.out.println(letter + "\n"+xOrO);
                        } else if((xOrO%2)==1 && getText().equals(" ") && win==false) {
                                letter="O";
                                xOrO=xOrO+1;
                                System.out.println(letter + "\n"+xOrO);
                        } // if user does click on a button that is already played, nothing will
happen

                        setText(letter); // place the x or the o on the actual board


                        for(int i=0; i<=7; i++){ // check for the winning combinations
                                if( buttons[winCombinations[i]
[0]].getText().equals(buttons[winCombinations[i][1]].getText()) &&
                                        buttons[winCombinations[i]
[1]].getText().equals(buttons[winCombinations[i][2]].getText()) &&
                                        buttons[winCombinations[i][0]].getText() != " "){//the
winning is true
                                        win = true;
                                }
                        }

                        if(win == true){ // if the game ends let the user know who wins and give
option to play again
                                again=JOptionPane.showConfirmDialog(null, letter + " wins the game!
Do you want to play again?",letter + "won!",JOptionPane.YES_NO_OPTION);

                        } else if(xOrO == 9 && win == false){//tie game, announce and ask if the user
want to play again
                                again=JOptionPane.showConfirmDialog(null, "The game was tie!  Do you
want to play again?","Tie game!",JOptionPane.YES_NO_OPTION);
                                win=true;
                        }

                        if(again==JOptionPane.YES_OPTION && win==true){ // if the user want to play
again clear all the button and start over
                                        clearButtons();
                                        win=false;
                        }
                        else if(again==JOptionPane.NO_OPTION){
                                System.exit(0); // exit game if the user do not want to play again
                        }




                }

        }

        public static void clearButtons(){

                for(int i=0; i<=8; i++){// clear all 8 buttons
                        buttons[i].setText(" ");
                }
                xOrO=0; // reset the count

        }

}
```