

Roll No:	907
Program No:	7
Name of Program:	Tic Tac Toe game
Objective:	To Design classic Multiplayer tic-tac-toe game

Source Code:

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace TicTacTeo
{
    public partial class frmMain : Form
    {
        Random myRandom;
        int turn;
        int count = 0;
        Boolean Winner = false;

        public frmMain()
        {
            InitializeComponent();
            myRandom = new Random();
            turn = myRandom.Next(2);
        }

        private void panel2_Paint(object sender, PaintEventArgs e)
        {
        }

        private void frmMain_Load(object sender, EventArgs e)
        {
            if (turn == 0)
            {
                radp1.Checked = true;
                radp1.ForeColor = Color.BlueViolet;
            }
            else
            {
                radp2.Checked = true;
                radp1.ForeColor = Color.BlueViolet;
            }
        }

        private void btn_Click(object sender, EventArgs e)
        {
            count++;
        }
    }
}

```

```

if (checkWinner("X"))
{
    MessageBox.Show("Player 1 wins");
    StopGame();
}

if(checkWinner("O"))
{
    MessageBox.Show("Player 2 wins");
    StopGame();
}

Button btn = (Button)sender;
if (turn == 0)
{
    btn.Text = "X";
    radp1.ForeColor = Color.Black;
    radp2.ForeColor = Color.BlueViolet;
    turn = 1;
}
else
{
    btn.Text = "O";
    radp2.ForeColor = Color.Black;
    radp1.ForeColor = Color.BlueViolet;
    radp1.Checked = true;
    turn = 0;
}
btn.Enabled = false;
if (Winner == false && count == 9)
{
    MessageBox.Show("its a draw");
    StopGame();
    btn.Enabled = true;
}
}

void StopGame()
{
    btn0.Enabled = false;
    btn1.Enabled = false;
    btn3.Enabled = false;
    btn4.Enabled = false;
    btn5.Enabled = false;
    btn6.Enabled = false;
    btn7.Enabled = false;
    btn8.Enabled = false;
    btn9.Enabled = false;
    if(MessageBox.Show("Do You want to restart the
game", "Continue", MessageBoxButtons.OKCancel)==DialogResult.OK)
    {
        ResetAll();
    }
    else
    {
        MessageBox.Show("Thank you for Playing!!!");
        this.Close();
    }
}

```

```

    }

}

void ResetAll()
{
    btn0.Enabled = true;
    btn1.Enabled = true;
    btn3.Enabled = true;
    btn4.Enabled = true;
    btn5.Enabled = true;
    btn6.Enabled = true;
    btn7.Enabled = true;
    btn8.Enabled = true;
    btn9.Enabled = true;
    btn0.Text = "";
    btn1.Text = "";
    btn3.Text = "";
    btn4.Text = "";
    btn5.Text = "";
    btn6.Text = "";
    btn7.Text = "";
    btn8.Text = "";
    btn9.Text = "";

    turn = myRandom.Next(2);
    if (turn == 0)
    {
        radp1.Checked = true;
        radp1.ForeColor = Color.BlueViolet;
    }
    else
    {
        radp2.Checked = true;
        radp1.ForeColor = Color.BlueViolet;
    }
}

```

```

bool checkWinner(String val)
{
    if(btn0.Text.Equals(val) && btn1.Text.Equals(val) && btn3.Text.Equals(val))
    {
        Winner = true;
        return true;
    }
    else if (btn4.Text.Equals(val) && btn5.Text.Equals(val) && btn6.Text.Equals(val))
    {
        Winner = true;
        return true;
    }
    else if (btn7.Text.Equals(val) && btn8.Text.Equals(val) && btn9.Text.Equals(val))
    {
        Winner = true;
        return true;
    }
}

```

```

    }
    else if (btn0.Text.Equals(val) && btn4.Text.Equals(val) && btn7.Text.Equals(val))
    {
        Winner = true;
        return true;
    }
    else if (btn1.Text.Equals(val) && btn5.Text.Equals(val) && btn8.Text.Equals(val))
    {
        Winner = true;
        return true;
    }
    else if (btn3.Text.Equals(val) && btn6.Text.Equals(val) && btn9.Text.Equals(val))
    {
        Winner = true;
        return true;
    }
    else if (btn0.Text.Equals(val) && btn5.Text.Equals(val) && btn9.Text.Equals(val))
    {
        Winner = true;
        return true;
    }
    else if (btn3.Text.Equals(val) && btn5.Text.Equals(val) && btn7.Text.Equals(val))
    {
        Winner = true;
        return true;
    }
    return false;
}

private void btnReset_Click(object sender, EventArgs e)
{
    if (MessageBox.Show("Do You want to restart the game", "Continue",
        MessageBoxButtons.OKCancel) == DialogResult.OK)
    {
        ResetAll();
    }
    else
    {
        MessageBox.Show("Thank you for Playing!!!");
        this.Close();
    }
}
}
}
}

```

OutPut:

