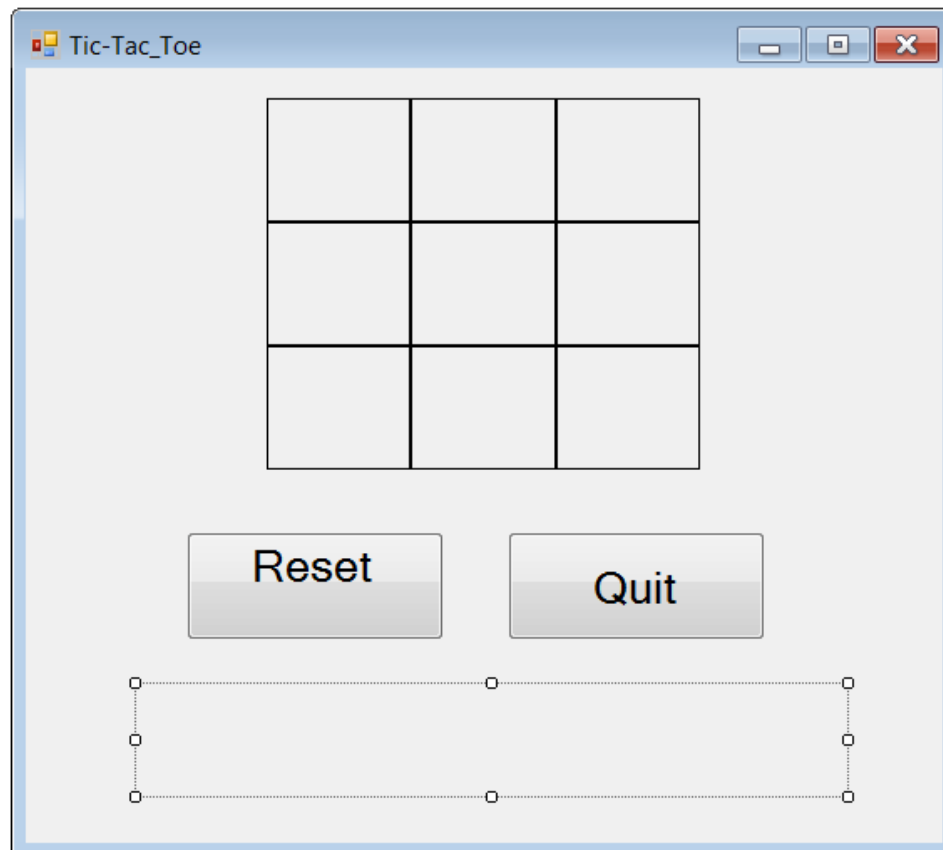


Name: **Session:**
Programming II
Making a Tic-Tac-Toe Game
Lab Exercise 3/31/2025

1. Start by building the following interface. The interface should contain 11 buttons and a label.

The controls should have the following names:

Name	Purpose
btnLeftUpper	To place an X or an O in the left upper square
btnMiddleUpper	To place an X or an O in the middle upper square
btnRightUpper	To place an X or an O in the right upper square
btnLeftMiddle	To place an X or an O in the left middle square
btnMiddleMiddle	To place an X or an O in the middle middle square
btnRightMiddle	To place an X or an O in the right middle square
btnLeftLower	To place an X or an O in the left lower square
btnMiddleLower	To place an X or an O in the middle lower square
btnRightLower	To place an X or an O in the right lower square
btnReset	To reset the game
btnQuit	To exit the game
lblWin	Displays winner message



2. Add the following global variables

```
bool player;  
bool disabled1;  
bool disabled2;  
bool disabled3;  
bool disabled4;  
bool disabled5;  
bool disabled6;  
bool disabled7;  
bool disabled8;  
bool disabled9;
```

3. Add the following code to the Form1_Load Sub

```
player = true;  
disabled1 = false;  
disabled2 = false;  
disabled3 = false;  
disabled4 = false;  
disabled5 = false;  
disabled6 = false;  
disabled7 = false;  
disabled8 = false;  
disabled9 = false;
```

4. Add the following two functions to your program. The first function will check to see if a player has won and the second will lock the cells after they have been selected.

```
private void checkWin()  
{  
    playerX_Win();  
    playerO_Win();  
}  
  
private void lockCells()  
{  
    btnLeftUpper.Enabled = false;  
    btnMiddleUpper.Enabled = false;  
    btnRightUpper.Enabled = false;  
    btnLeftMiddle.Enabled = false;  
    btnMiddleMiddle.Enabled = false;  
    btnRightMiddle.Enabled = false;  
    btnLeftLower.Enabled = false;  
    btnMiddleLower.Enabled = false;  
    btnRightLower.Enabled = false;  
}
```

5. Now we will add two functions that will check if playerX or playerO has won.

```
private void playerX_Win()
{
    if (btnLeftUpper.Text == "X" && btnMiddleUpper.Text == "X" && btnRightUpper.Text == "X")
    {
        lblWin.Text = "Player X Wins";
        lockCells();
    }

    if (btnLeftLower.Text == "X" && btnMiddleLower.Text == "X" && btnRightLower.Text == "X")
    {
        lblWin.Text = "Player X Wins";
        lockCells();
    }

    if (btnLeftMiddle.Text == "X" && btnMiddleMiddle.Text == "X" && btnRightMiddle.Text == "X")
    {
        lblWin.Text = "Player X Wins";
        lockCells();
    }

    if (btnLeftUpper.Text == "X" && btnLeftMiddle.Text == "X" && btnLeftLower.Text == "X")
    {
        lblWin.Text = "Player X Wins";
        lockCells();
    }

    if (btnMiddleUpper.Text == "X" && btnMiddleMiddle.Text == "X" && btnMiddleLower.Text == "X")
    {
        lblWin.Text = "Player X Wins";
        lockCells();
    }

    if (btnRightUpper.Text == "X" && btnRightMiddle.Text == "X" && btnRightLower.Text == "X")
    {
        lblWin.Text = "Player X Wins";
        lockCells();
    }

    if (btnLeftUpper.Text == "X" && btnMiddleMiddle.Text == "X" && btnRightLower.Text == "X")
    {
        lblWin.Text = "Player X Wins";
        lockCells();
    }

    if (btnRightUpper.Text == "X" && btnMiddleMiddle.Text == "X" && btnLeftLower.Text == "X")
    {
        lblWin.Text = "Player X Wins";
        lockCells();
    }
}
```

Note: playerO_Win() is similar

6. Now we will write code for the 9 cell button click events:

```
private void btnLeftUpper_Click(object sender, EventArgs e)
{
    if (player)
    {
        if (!disabled1)
        {
            btnLeftUpper.Text = "X";
            player = !player;
            disabled1 = true;
        }
    }
    else
    {
        if (!disabled1)
        {
            btnLeftUpper.Text = "O";
            player = !player;
            disabled1 = true;
        }
    }
    checkWin();
}
```

7. The Click event handlers for btnMiddleUpper, btnRightUpper, btnLeftMiddle, btnMiddleMiddle, btnRightMiddle, btnLeftLower, btnMiddleLower, and btnRightLower are similar.

8. Next we will add the code for the btnReset and btnQuit click events.

```
private void btnReset_Click(object sender, EventArgs e)
{
    player = true;
    disabled1 = false;
    disabled2 = false;
    disabled3 = false;
    disabled4 = false;
    disabled5 = false;
    disabled6 = false;
    disabled7 = false;
    disabled8 = false;
    disabled9 = false;
    btnLeftUpper.Enabled = true;
    btnMiddleUpper.Enabled = true;
    btnRightUpper.Enabled = true;
    btnLeftMiddle.Enabled = true;
    btnMiddleMiddle.Enabled = true;
    btnRightMiddle.Enabled = true;
}
```

```
        btnLeftLower.Enabled = true;
        btnMiddleLower.Enabled = true;
        btnRightLower.Enabled = true;
        btnLeftUpper.Text = "";
        btnMiddleUpper.Text = "";
        btnRightUpper.Text = "";
        btnLeftMiddle.Text = "";
        btnMiddleMiddle.Text = "";
        btnRightMiddle.Text = "";
        btnLeftLower.Text = "";
        btnMiddleLower.Text = "";
        btnRightLower.Text = "";
        lblWin.Text = "";
    }
```

9. Enjoy the game....

Submit a screen shot of a game you played.