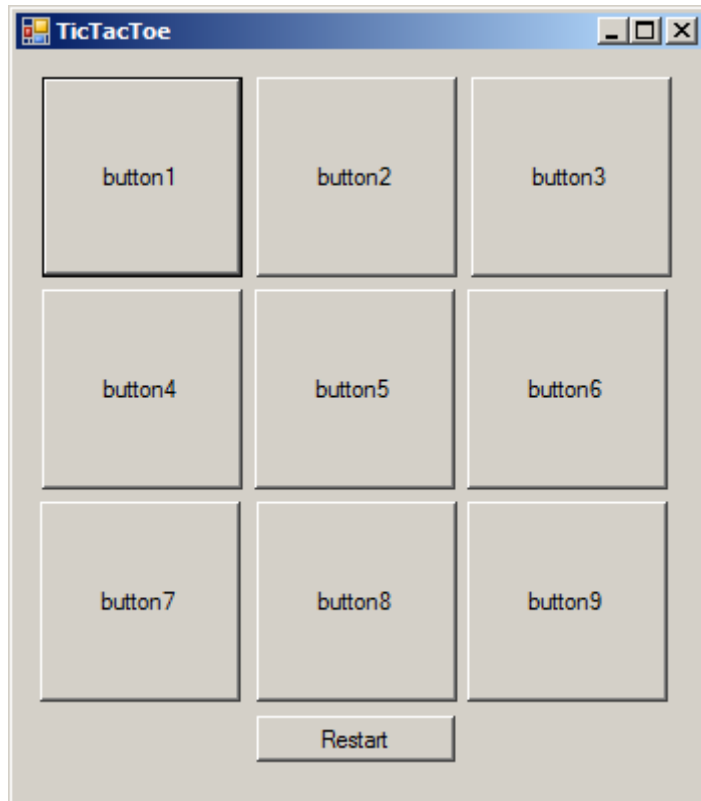


## TicTacToe Spiel

Erstellen Sie eine Windows Forms Anwendung mit folgendem Aussehen:



Daten und Namen in der Form:

### Main Form

#### Size

Breite: 350  
Höhe: 400

#### Text

TicTacToe

### Spiel-Buttons (button1 bis button9)

#### Size

Breite: 100  
Höhe: 100

### Restart Button (restartButton)

#### Size

Breite: 100  
Höhe: 23

#### Text

Restart

Erstellen Sie ein Array vom Typ Button, welchem alle Spiel-Buttons hinzugefügt werden.

Fügen Sie anschließend folgenden Quelltext ein. Im Konstruktor des Forms nach `InitializeComponent()`

```
public Form1()
{
    InitializeComponent();
```

Eine Schleife, welche über das Button-Array läuft und in welcher die folgenden Zeilen ausgeführt werden:

```
// Text leeren
buttonArray[i].Text = "";

// Font setzen
buttonArray[i].Font = new System.Drawing.Font("Microsoft Sans Serif",
40F);

// Event setzen
buttonArray[i].Click += new System.EventHandler(ClickHandler);
```

Legen Sie die Methode `ClickHandler()` wie folgt an:

```
private void ClickHandler(object sender, System.EventArgs e)
{
    Button angeklickterButton = (Button)sender;
}
```

In der Methode `ClickHandler()` sind mehrere Zustände zu prüfen:

- Ob das Spiel schon beendet ist
- Ob das angeklickte Feld noch leer ist (also kein x oder o ist) (Eigenschaft `Text`)
- Wenn es leer ist, im Wechsel ein x oder ein o als Text setzen (Eigenschaft `Text`)
- Überprüfen ob ein Spieler gewonnen hat

Meldungen an den Nutzer können mittels `MessageBox.Show("Text")` ausgegeben werden.

Der Reset-Button muss alle nötigen Werte zurücksetzen, um das Spiel neustarten zu können.