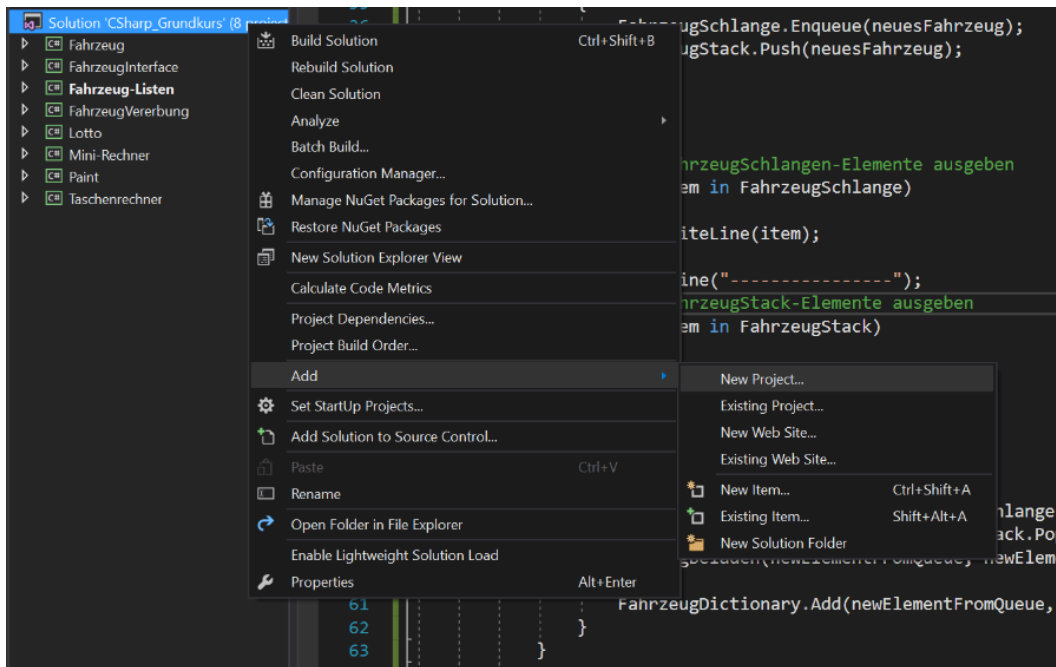


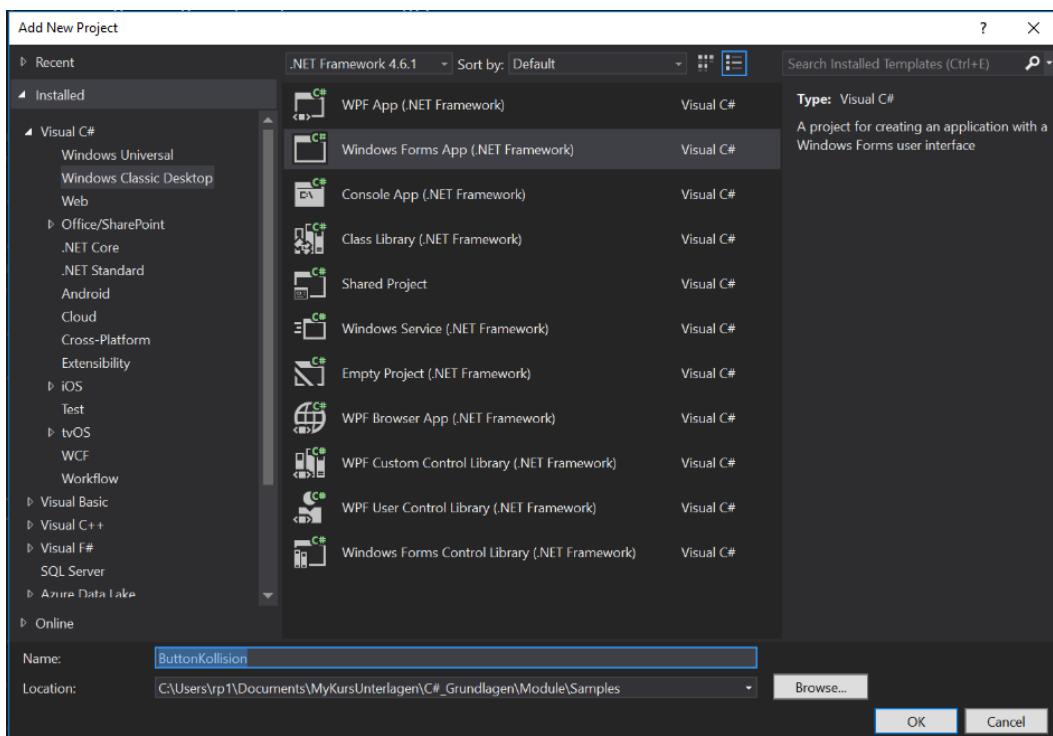
Lab09: WindowsForm-Events

1. Vorbereitung

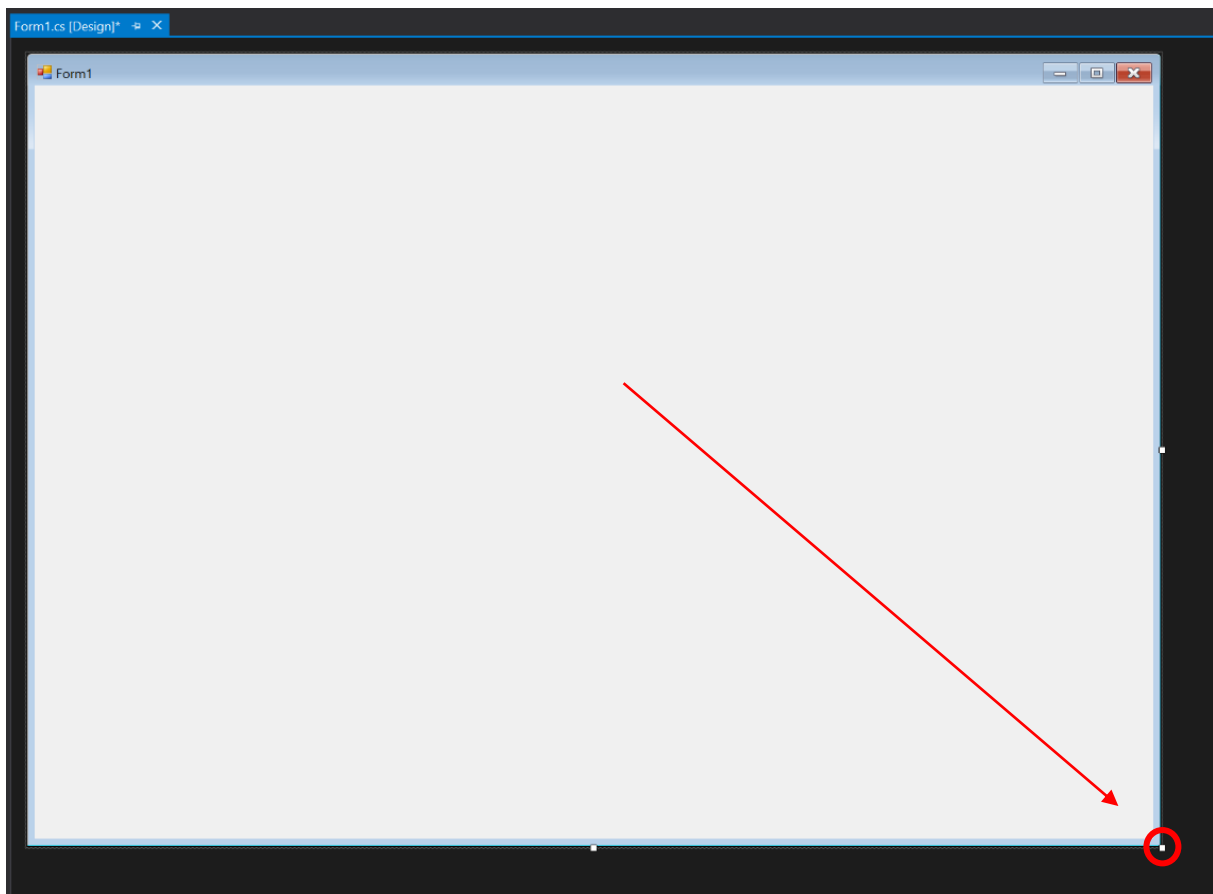
Fügen Sie Ihrer Solution (Projektmappe) ein neues Projekt hinzu mittels Rechts-Klick auf Solution und Add → New Project.



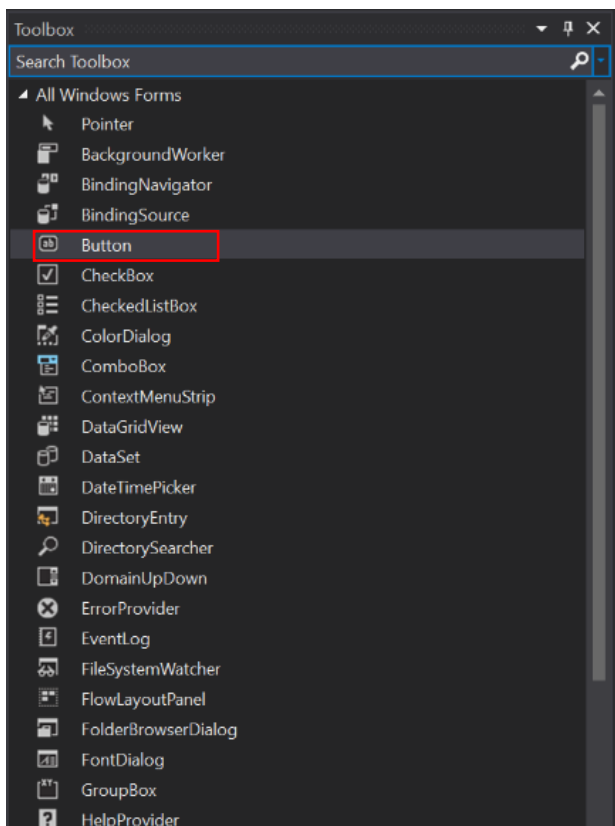
Wählen Sie im Dialog die Projektart „Windows Forms App (.NET Framework)“ aus und geben Sie dem Projekt den Namen „ButtonKollision“.



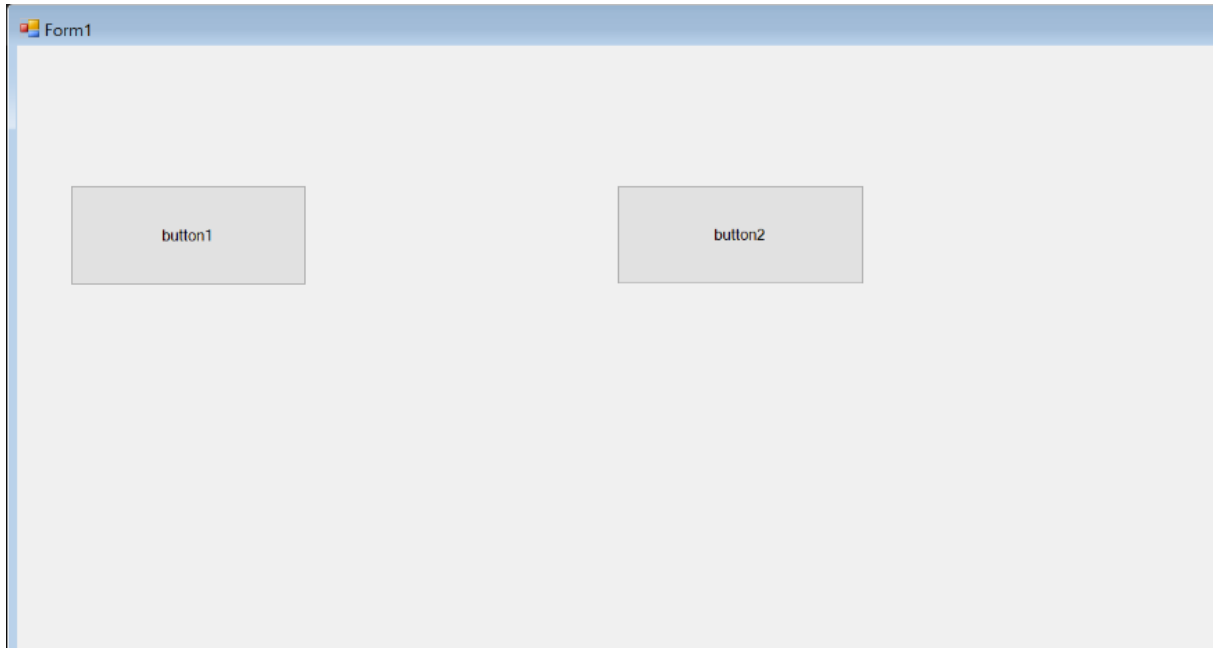
Jetzt Sehen Sie den Formular-Designer. Ziehen Sie das Formular möglichst groß.



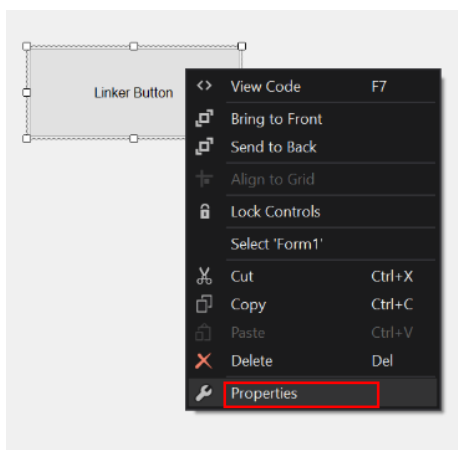
Öffnen Sie nun übers Menü View → Toolbox oder die Tastenkombination Strg+Alt+X die Toolbox. Darin Öffnen Sie die Sektion „All Windows Forms“ und wählen dort das Element Button aus.



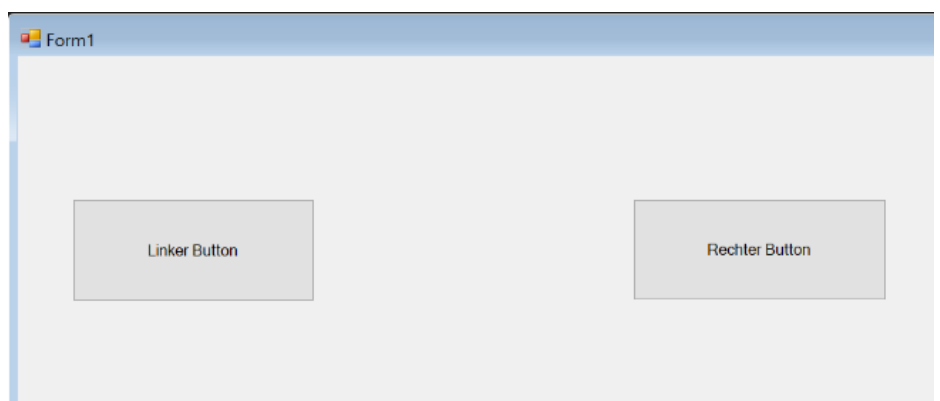
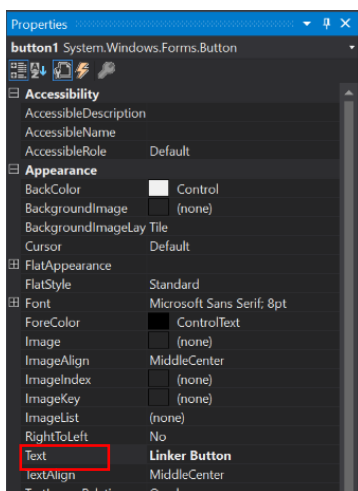
Platzieren Sie nun 2 Buttons auf dem Formular indem Sie mit der Maus ein Rechteck ziehen. Bevor Sie den zweiten Button ziehen können, müssen Sie zuvor erneut den Eintrag in der Toolbox auswählen oder Sie kopieren den Button mittels Strg-C und Strg-V.



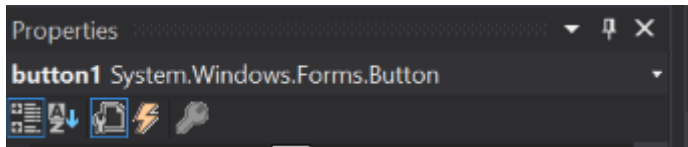
Klicken Sie mit der rechten Maustaste und wählen Sie Properties.



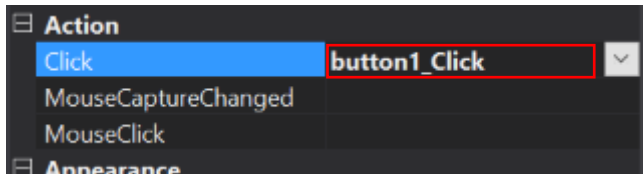
Das Properties-Fenster für diesen Button öffnet sich, darin ändern Sie mittels der Eigenschaft „Text“ die Beschriftung des Buttons zu „Linker Button“ bzw. „Rechter Button“ für den rechten Button.



In der Kopfzeile des Properties-Windows sehen Sie verschiedene Buttons. Klicken Sie auf das Blitz-Symbol um die Events des ausgewählten Buttons zu sehen.

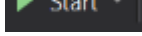


Klicken Sie doppelt in das Feld fürs Click-Event:



Nun sollte im Texteditor automatisch die Funktion `button1_Click` angelegt werden. Darin können Sie nun den Code schreiben, der beim Klick auf den Button ausgelöst werden soll. Schreiben Sie folgenden Code hinein:

```
private void button1_Click(object sender, EventArgs e)
{
    button1.Left++;
}
```

Kompilieren Sie das Programm über Klick auf  und achten sie darauf dass in der Auswahlbox links daneben das richtige Programm ausgewählt ist.

Der linke Button sollte sich nun bei Klick auf ihn schrittweise nach rechts bewegen. Implementieren Sie auch für den rechten Button das Klick-Event und lassen Sie ihn allerdings nach links wandern.

```
private void button2_Click(object sender, EventArgs e)
{
    button2.Left--;
}
```

Die Eigenschaften `Left`, `Right`, `Top` und `Bottom` steuern die Position eines Elementes, mittels `Width` und `Height` definieren Sie Breite und Höhe.

2. Aufgabe

Erzeugen Sie einen dritten Button und weisen Sie ihm code-seitig für sein Click-Event die EventHandler-Funktionen der beiden anderen Buttons zu (`button1_Click` und `button2_Click`), so dass beim Klick auf den dritten Button die beiden ersten Buttons sich gleichzeitig aufeinander zubewegen.

Bauen Sie weiterhin eine Prüfung ein ob die beiden Buttons miteinander kollidieren und geben Sie in dem Fall eine Meldung mittels der Methode `MessageBox.Show()` aus.

Viel Erfolg!