

Hochschule Harz	FB Automatisierung und Informatik
1. Vorleungsübung Dialog mit IO	Grafische Nutzerschnittstellen mit .net Thema: Erstellen eines dialogbasierten Programms mit Ein- und Ausgabe

Versuchsziele

- Konzepte der .net-Programmierung am Beispiel eines Dialogfenster anwenden können
- Kenntnisse in der Implementierung von GUI-Elementen und Befehlen zum Einlesen von ASCII-Dateien
- Dynamisches Erzeugen von GUI-Elementen

Einlesen einer ASCII-Datei

Überblick

Erstellen eines Dialogprogramms mit C# und .net zum Einlesen von ASCII-Dateien. Das Fenster hat eine Schalterleiste mit einem Registern. Im Einzelnen sollen folgende Ziele vermittelt werden:

- Verwendung von eines toolStripContainers
- TabControl verwenden
- Laden einer Ascii-Datei mit StreamReader
- Zeitmessung bei den unterschiedlichen Varianten
- Dynamisches Eintragen neuer Komponenten
- Anzahl der Zeilen 148 Zeilen

Aufgaben

1. Aufgabe:

- Erstellen einer Form in C#
- Einbau eines MenüContainers
 - Eintragen des Hauptmenüs „Datei“, Name MainFile
 - Eintragen eines Eintrags „Schließen, Name: MnClose
- Einbau eines toolStripContainers
- Dock auf DockStyle=Fill setzen
- Einfügen eines toolStrip
- Einfügen dreier Schalter
 - Name: BnRead1
 - Name: BnRead2
 - Name: BnRead3
- Einfügen eines Separators
- Einfügen eines Labels
 - Name: LTimer
- Einfügen eines Separators
- Einfügen eines weiteren Schalters
 - Name: BnMemo
- Einfügen eines TabControl

- Name: MainTab
- Löschen des zweiten Registers
- Einfügen eines Editors in das erste Registers
 - Name: Editor
 - Scrollbar auf both setzen
 - wordwrap ausschalten
- Doppelklick in Form, Erstellen load_form
- Eintragen:
 - MainTab.Dock = DockStyle.Fill;
 - Editor.Dock = DockStyle.Fill;
 - LTime.Text = "0";
- Testen

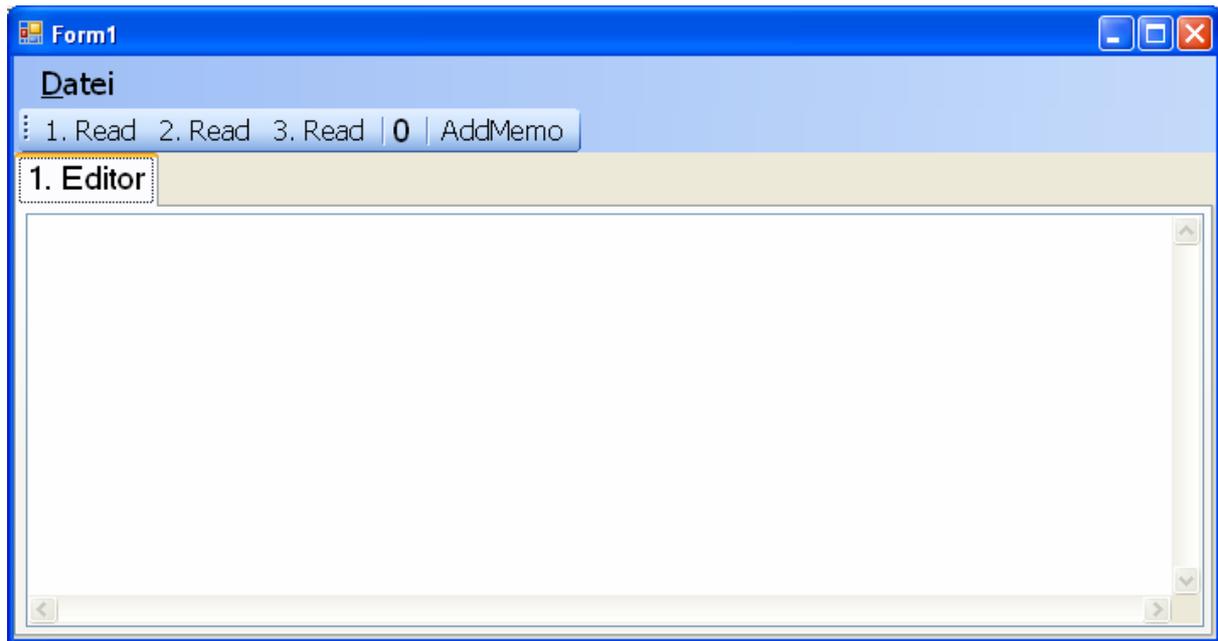


Abbildung 1 Anfangsdialog

2. Aufgabe: BnRead1_Click

- Erstellen der OnClick-Methode vom Schalter „BnRead1“
- Aufrufen eines Open-Dialogs
- Filter *.txt
- Öffnen der Datei
- Aufruf der Methode read1(string sFilename)
- Zeit 1 setzen
- Zeilenweise Einlesen
- nach **jedem** Lesen wird die Zeile in den Editor eingefügt
- Zeit 2 setzen
- Zeitspanne mit LTimer ausgeben

3. Aufgabe: BnRead2_Click

- Erstellen der OnClick-Methode vom Schalter „BnRead1“
- Aufrufen eines Open-Dialogs
- Filter *.txt
- Öffnen der Datei
- Aufruf der Methode read1(string sFilename)

- Zeit 1 setzen
- Zeilenweise Einlesen
- ~~nach jedem Lesen wird die Zeile in den Editor eingefügt~~
- Zeit 2 setzen
- Zeitspanne mit LTimer ausgeben

4. Aufgabe: BnRead3_Click

- Erstellen der OnClick-Methode vom Schalter „BnRead1“
- Aufrufen eines Open-Dialogs
- Filter *.txt
- Öffnen der Datei
- Aufruf der Methode read1(string sFilename)
- Zeit 1 setzen
- Zeilenweise Einlesen
- ~~nach jedem Lesen wird die Zeile in den Editor eingefügt~~
- Zeit 2 setzen
- Zeitspanne mit LTimer ausgeben

5. Aufgabe: Delegate-Fuinktion einbauen

- Erstellen einer Delegate-Funktion
 - public delegate void Read_Delegate(string sFilename);
- Einbauen in die drei Read-Click-Methoden

Beispiel:

```
private void BnRead1_Click(object sender, EventArgs e)
{
    OpenFilename(Read1);
}

private void OpenFilename(Read_Delegate Read)
{
    // Aktion
}
```

6. Aufgabe: BnMemo_Click

- Erstellen der OnClick-Methode vom Schalter „BnMemo“
- Erstellen eines neuen Registers
- Erstellen eines neuen Editors
- Setzen der Eigenschaften (Scrollbars, wordwrap etc.)
- Setzen der Caption für das neue Register
- **Setzen des Namens für den Editor**
- Einfügen eines Editors in das neue Register

7. Aufgabe: BnEinst_Click

- Erstellen eines neuen Frames,
 - Name: FEinst
 - Schalter BnOk erhält den DialogResult Ok
 - Schalter BnEsc erhält den DialogResult Cancel
- Einbau der get-Methoden für die Attribute
 - `public int getSize() // Cast nach int`
 - `public int getColor() 0,1,2`
- Eintragen des Hauptmenüs „Hilfe“, Name MainHelp
- Eintragen eines Eintrags „Einstellungen, Name: MnEinst
- Erstellen der Click-Methode
- Aufrufen des Einstellungs-Dialog
- Abfragen des Ok-Schalter
- Setzen der Werte im Haupt-Editor

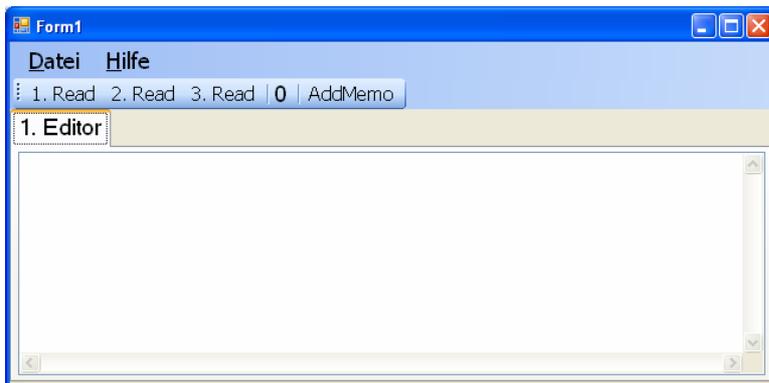


Abbildung 2 Vollständiges Programm

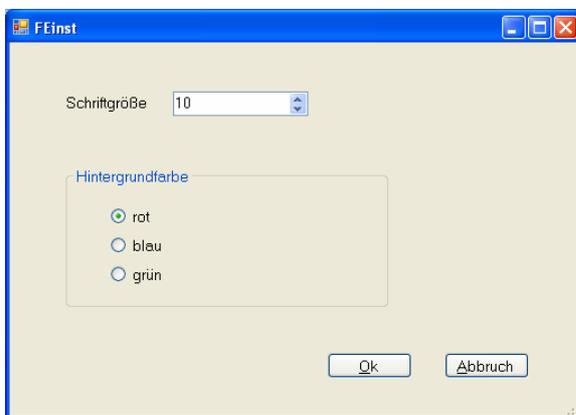


Abbildung 3 Einstellungsdialog

Der NumericUpDown hat den Namen „spSize“

Minimalwert: 8

Maximalwert: 30

Die RadioButton haben die Namen: rbRed rbBlue rbGreen

Setzen eines Fonts

```
Editor.Font = new System.Drawing.Font("Tahoma", size,  
    System.Drawing.FontStyle.Regular,  
    System.Drawing.GraphicsUnit.Point, ((byte)0));
```

Anhang

Öffnen einer Datei:

```
OpenFileDialog openFileDialog1 = new OpenFileDialog();
openFileDialog1.Filter =
    "pdf files (*.pdf)|*.pdf|All files (*.*)|*.*";
openFileDialog1.FilterIndex = 2;// zählt von 1 !
openFileDialog1.DefaultExt = ".txt";
openFileDialog1.InitialDirectory="c:\\daten\\0";
openFileDialog1.Multiselect=false;
openFileDialog1.RestoreDirectory = true;

if (openFileDialog1.ShowDialog() == DialogResult.OK){
    string sFile=openFileDialog1.FileName.ToString()
}
```

Speichern einer Datei:

```
SaveFileDialog saveFileDialog1 = new SaveFileDialog();
saveFileDialog1.Filter =
    "pdf files (*.pdf)|*.pdf|All files (*.*)|*.*";
saveFileDialog1.FilterIndex = 1;// zählt von 1 !
saveFileDialog1.DefaultExt = ".txt";
saveFileDialog1.InitialDirectory = "c:\\daten";
saveFileDialog1.RestoreDirectory = true;
if (saveFileDialog1.ShowDialog() == DialogResult.OK)
{
    string sFile = saveFileDialog1.FileName.ToString();
    MessageBox.Show(sFile, Application.ProductName);
}
```

Lesen einer Datei

```
FileStream inFile = new FileStream(sFilename, FileMode.Open);

StreamReader inStream = new StreamReader(inFile);
while (! inStream.EndOfStream) {
    String value = inStream.ReadLine();
}

inStream.Close();
inFile.Close
```

Zeitmessung

```
DateTime Time1 = DateTime.Now;
// Aktion
DateTime Time2 = DateTime.Now;
TimeSpan duration1 = Time2 - Time1;
```