

„Euro-Rechner“

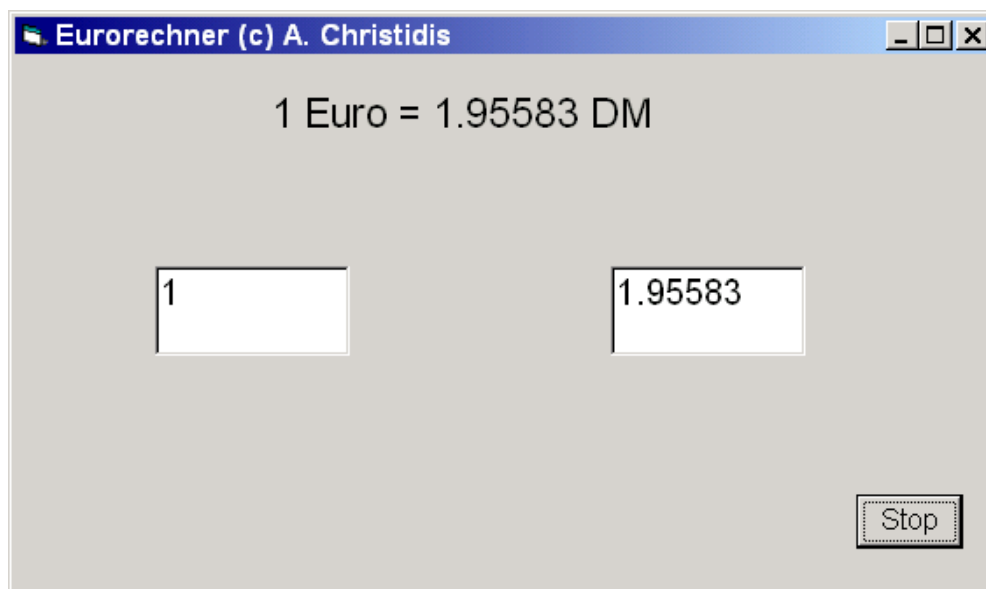
Implementieren Sie einen Währungsumrechner für DM und Euro in Anlehnung an das u.a. Muster. Gehen Sie dabei wie folgt vor:

- Erstellen Sie zunächst ein Programm, das nur in einer Richtung (Euro zu DM) rechnet. Verwenden Sie dazu ein Bezeichnungsfeld (Label) und zwei Textfelder, und nutzen Sie das von VB zur Verfügung gestellte Ereignis `Change()`. Führen Sie Ihr Programm vor, und sichern Sie diesen Stand als eigenständiges Projekt (z.B. Eu2DM).
- Erzeugen Sie eine Kopie des Projektes (z.B. Eu2DMvv) und versuchen Sie, die Konvertierung in beide Richtungen laufen zu lassen. Was bewirkt die ereignisorientierte Funktionsweise des Programms?
- Nach Fertigstellung erstellen Sie ein ausführbares Programm (Eu2DMvv.exe) und spielen Sie damit. Was beobachten Sie bzgl. der Eingabe und Ausgabe von Nachkommastellen? Wieviel Euro sind 1 Pfennig?

Hinweise:

- Nutzen Sie die VB-Funktion `Val(ZEICHENKETTE)` zur Interpretation einer Zeichenkette als Zahl.
- Üben Sie (erst in der erweiterten Version!) mit der `Format$`-Anweisung (s. entsprechende Notiz) und treffen Sie eine (begründete) Entscheidung, die der Umrechnungsgenauigkeit gerecht wird!

1 Visueller Entwurf der Bedienoberfläche



2 Zuweisung der Objekteigenschaften

Auf jeweiliges Objekt klicken und Eigenschaften zuweisen:

	Frm	Lbl	Txt1	Txt2	Cmd1
Name	frmEuro	lblKurs	txtEuro	txtDM	cmdStop
Alignment		2 – Zentriert			
Caption	Eurorechner (c) <Ihr Name>	1 Euro = 1.95583 DM			Stop
Font		Arial Standard 14	Arial Standard 12	Arial Standard 12	Arial Standard 10
Text			1	1,95583	

3 Verknüpfung Objekte / Ereignisse (Erster Versuch)

```
Option Explicit
Private Sub txtEuro_Change()
txtDM.Text = Val(txtEuro.Text) * 1.95583
End Sub

Private Sub cmdStop_Click()
Unload frmEuro
End Sub
```

3.1 Weitere Fragen

- Erweitern Sie nun selbständig das Programm bis zur o.a. vollen Funktionalität!
Hinweis: Speichern Sie regelmäßig den Stand Ihrer Arbeit, weil je nach Handhabung Rechner-Abstürze mit Datenverlust möglich sind!
- Lassen Sie Ihr Programm mit dem Debugger schrittweise laufen!
- Setzen Sie Haltepunkte an jeden Unterprogramm-Eintritt und lassen Sie die Ausführung Ihres Programms an diesen Stellen anhalten.
- Fragen Sie die aktuellen Werte der Variablen ab, wenn die Programm-Ausführung unterbrochen wird. Welches sind die Gültigkeitsbereiche Ihrer Variablen?
- Lassen Sie das System einzelne Variablen überwachen und experimentieren Sie, indem Sie die Variablen-Werte auf willkürliche Werte setzen.
- Schicken Sie bitte schließlich die Dateien dieses Projektes (*.exe, *.frm, *.vbw, *.vbp) gezippt an A.Christidis@mni.fh-giessen.de