 Friedrich-Ebert-Schule Esslingen	<b>µC-Steuerungen mit Visual Basic</b>	Name:
<b>2.3</b>	<b>Begriffe der Objektorientierten Programmierung</b>	Datum:

In VB 2005/2008 ist die Objektorientierte Programmierung (OOP) konsequent umgesetzt worden. Auf die Besonderheiten der OOP soll hier nicht eingegangen werden, jedoch sind einige Begrifflichkeiten wichtig, damit sich bestimmte Vorgehensweisen auch dem „klassischen“ Programmierer erschließen.

## Begriffe der OOP am Beispiel der SerialPort-Klasse

- Eine **Klasse** beschreibt die Eigenschaften eines Objekts. Sie ist damit der Bauplan für ein Objekt. Einige tausend Klassen sind in den .Net-Klassenbibliotheken (.Net-Framework) definiert.
- **Objekte** sind Realisierungen von Klassen und werden in Objektvariablen gespeichert. Man bezeichnet ein Objekt daher auch als Instanz einer Klasse.  
*Bsp.:* Eine Objektvariable wird mit dem **Konstruktor** *New* von einer Klasse abgeleitet.  

```
Dim ComPort As New System.IO.Ports.SerialPort
```
- **Namensräume:** Zur Übersichtlichkeit werden .Net-Klassen in funktionale Gruppen eingeteilt. Bei einem Bezug auf eine bestimmte Klasse werden die Gruppen und Untergruppen mit einem Punkt getrennt.  
*Bsp.:* Der voll qualifizierte Klassenname der SerialPort-Klasse lautet:  

```
System.IO.Ports.SerialPort
```

 Da Klassennamen ziemlich lang sein können, kürzt man Namensräume oft mit der Imports-Anweisung ab. Sie muss immer am Anfang der VB-Datei stehen!  
*Bsp.:* Abkürzung der Deklaration der SerialPort-Objektvariablen  

```
Imports System.IO.Ports
```

 Damit kann die Objektvariable wie folgt deklariert werden.  

```
Dim ComPort As New SerialPort
```
- **Methoden** sind besondere Funktionen, die auf Objekte angewendet werden. Mit Methoden werden Daten eines Objekts ermittelt oder das Objekt manipuliert.  
*Bsp.:* Die Methode *Open()* des ComPort-Objekts öffnet den seriellen Port.  

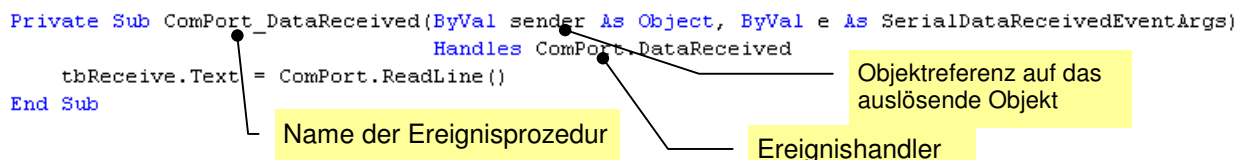
```
ComPort.Open()
```
- **Eigenschaften** sind die „Variablen“ eines Objekts. Sie können gelesen und meistens auch verändert werden.  
*Bsp.:* Die Eigenschaft *Baudrate* des ComPort-Objekts wird auf 9600Baud gesetzt.  

```
ComPort.BaudRate = 9600
```

 Die *IsOpen*-Eigenschaft des ComPort-Objekts wird abgefragt!  

```
If Not ComPort.IsOpen Then ComPort.Open()
```
- **Ereignisse** treten auf, wenn ein Objekt in einer bestimmten Weise verändert wird. Ereignisse lösen **Ereignisprozeduren** aus, in denen der Programmierer die Reaktion auf das Ereignis definiert. (Vergleiche: Interruptprogrammierung)  
*Bsp.:* Das *ComPort.DataReceived*-Ereignis wird ausgelöst, wenn ein Zeichen von der seriellen Schnittstelle empfangen wird.

```
Private Sub ComPort_DataReceived(ByVal sender As Object, ByVal e As SerialDataReceivedEventArgs)
    Handles ComPort.DataReceived
    tbReceive.Text = ComPort.ReadLine()
End Sub
```



Der Name der Ereignisprozedur kann (im Gegensatz zu älteren VB-Versionen) beliebig gewählt werden. Einer Ereignisprozedur dürfen auch mehrere Ereignishandler zugeordnet werden.