

Zusammenfassung zum File System Watcher

Event-Handling im .NET Framework



```
BtnBeispiel.Click += new SystemEventHandler(BtnBeispiel_Clicked);
```

Parameter von Handler-Methoden: Rückgabe : void

Parameter: object sender, EventArgs e

Events eines Forms: - Load beim Laden des Forms
- FormClosing beim Schließen des forms

MessageBox.Show(...)

Parameter: Text, Überschrift, MessageBoxButtons, MessageBoxIcon

File System Watcher ...

Überwacht Verzeichnisse auf Änderungen mit Hilfe von Events

- Changed
- Rename
- Created
- Deleted

Exception Objekte:

Properties:

- Message
- Source
- HelpLink

Methoden:

- ToString()
- Konstruktor mit String Parameter

Structured Exception Handling

```
try
{
    Code ...                                // try-Block mit Code (Programmlogik)

    throw ...                               // ggf. eigene Exceptions werfen (throw)
}
catch (Exception ex)
{
    Fehlerbehandlung ... //catch-Block fängt die Exception anhand
                        // des Datentyps (Fehlerlogik)
}
finally
{
    ...                                     // ggf. finally-Block: wird immer ausgeführt
                                        // -> „Aufräumarbeiten“
}
```

Vorteile:

- Trennung von Logik und Fehlerbehandlung => Übersichtlichkeit
- Genaue Informationen über den aufgetretenen Fehler
- Mehrere catch-Blöcke möglich => Reaktion auf unterschiedliche Fehler

Eigene Exception Objekte:

... werden von System.Exception (oder einer davon abgeleiteten Klasse) abgeleitet.

-> Konstruktor der Basisklasse für die Fehlermeldung nutzen (Message Property)

```
public class MyException : Exception
{
    public MyException() : base („Fehlermeldung ...“)
    {
        ...
    }
    ...
}
```