



# Software Development

## Quellenangaben:

- Jürgen Kotz, München, Visual C# 2019, Carl Hanser Verlag München
- Michael Bonacina, C# Programmieren für Einsteiger (2. Aufl.), BMU Verlag
- <https://docs.microsoft.com/de-de/dotnet/csharp/programming-guide>

# File I/O Handling

# Zugriff auf das Dateisystem

- Für den Zugriff auf das Dateisystem gibt es fertig implementierte Klassen im .NET Framework (System.IO-Namespace)

Klasse	Beschreibung
Directory	Die statischen Methoden erlauben das Erstellen, Verschieben und Benennen von Verzeichnissen und Unterverzeichnissen
DirectoryInfo	Ähnelt der Directory-Klasse, enthält aber nur Instanzmethoden
Path	Die statischen Methoden erlauben die plattformübergreifende Arbeit mit Verzeichnissen
File	Die statischen Methoden erlauben das Erzeugen, Kopieren, Löschen, Verschieben und Öffnen von Dateien
FileInfo	Ähnelt der File-Klasse, aber nur Instanzmethoden
FileSystemInfo	Basisklasse für DirectoryInfo- und FileInfo-Objekte
DriveInfo	Liefert Laufwerksinformationen

# Statische vs Instanzen-Klassen

- Bei statischen Klassen muss bei jedem Methodenaufruf der Dateiname oder der Verzeichnispfad übergeben werden
- Bei Instanzen-Klassen FileInfo und DirectoryInfo kann der Datei- oder Verzeichnisname im Konstruktor einmalig spezifiziert werden
- Instanzen-Klassen haben bessere Performance da z.B. die Sicherheitsprüfung (Lese/Schreib-Berechtigung) nur ein einziges mal geprüft wird

# Statische vs Instanzen-Klassen

## Beispiel

### C #

Mit statischer Klasse:

```
string creationTime = File.GetCreationTime("Example.txt").ToString();  
string lastAccessTime = File.GetLastAccessTime("Example.txt").ToString();
```

Oder mit Instanzen-Klasse:

```
FileInfo myFile = new FileInfo("Example.txt");  
string creationTime = myFile.CreationTime.ToString();  
string lastAccessTime = myFile.LastAccessTime.ToString();
```

# Schreibweise

## Klasse Directory/DirectoryInfo

### C #

Mit statischer Klasse:

```
// Erstellen eines Directory  
Directory.CreateDirectory(@"c:\temp\Testverzeichnis");
```

Mit Instanzen-Klasse:

```
// Erstellen eines Directory mit Instanz-Methoden  
var directory = new DirectoryInfo(@"c:\temp\Testverzeichnis2");  
directory.Create();
```

# Methoden der Directory-Klasse

Die folgende Tabelle gibt eine Übersicht über wichtige statische Methoden:

Methode	Beschreibung
CreateDirectory	erzeugt ein Verzeichnis oder ein Unterverzeichnis
Delete	löscht ein Verzeichnis
Exists	überprüft die Existenz eines Verzeichnisses
GetCreationTime	liefert den Zeitpunkt der Erstellung
GetDirectories	liefert die Namen aller Unterverzeichnisse eines bestimmten Ordners
GetFiles	liefert alle Dateinamen eines bestimmten Ordners
GetParent	liefert den Namen des übergeordneten Verzeichnisses
Move	verschiebt ein Verzeichnis inklusive Dateien in ein anderes Verzeichnis
SetCreationTime	legt den Zeitpunkt der Erstellung eines Verzeichnisses fest



# Methoden eines DirectoryInfo-Objekts

in der folgenden Tabelle wurden einige wichtige Methoden weggelassen, da es diese mit gleicher bzw. ähnlicher Bedeutung bereits in der Directory-Klasse gibt

Methode	Beschreibung
CreateSubdirectory	erstellt ein oder mehrere Unterverzeichnis(se) im angegebenen Pfad
GetFileSystemInfos	ruft ein Array von FileSystemInfo-Objekten mit Dateien und Unterverzeichnissen ab
MoveTo	verschiebt ein DirectoryInfo-Objekt nebst Inhalt in einen neuen Pfad

# Eigenschaften eines DirectoryInfo-Objekts

Eigenschaft	Beschreibung
CreationTime	liest oder schreibt den Erstellungszeitpunkt des Verzeichnisses
Exists	liest einen Wert, der angibt, ob das Verzeichnis vorhanden ist
FullName	liest den vollständigen Pfad des Verzeichnisses
LastAccessTime	liest oder schreibt den Zeitpunkt des letzten Zugriffs auf das aktuelle Verzeichnis
LastWriteTime	liest oder schreibt den Zeitpunkt des letzten Schreibzugriffs auf das aktuelle Verzeichnis
Parent	liest das übergeordnete Verzeichnis eines angegebenen Unterverzeichnisses
Root	liest das Stammverzeichnis eines Pfads

# Methoden der File-Klasse

Die folgende Tabelle gibt eine Übersicht über wichtige statische Methoden:

Methode	Typ	Beschreibung
AppendAllText	void	öffnet die Datei und hängt eine Zeichenfolge an
AppendText	StreamWriter	hängt Text an eine vorhandene Datei an
Copy	void	kopiert eine Datei an einen anderen Speicherort
Create	FileStream	erzeugt eine Datei in einem bestimmten Pfad
CreateText	StreamWriter	erzeugt und öffnet eine Textdatei
Delete	void	löscht eine Datei
Exists	Boolean	prüft die Existenz einer Datei
GetAttributes	FileAttributes	liefert die Dateiattribute
GetCreationTime GetLastAccessTime GetLastWriteTime	DateTime	liefert Zeitpunkt der Erstellung, des letzten Zugriffs, des letzten Schreibzugriffs

# Methoden der File-Klasse

## Fortsetzung

Die folgende Tabelle gibt eine Übersicht über wichtige statische Methoden:

Methode	Typ	Beschreibung
Move	void	verschiebt eine Datei oder benennt sie um
Open	FileStream	öffnet eine Datei
OpenRead	FileStream	öffnet eine Datei zum Lesen
OpenText	StreamReader	öffnet eine Textdatei zum Lesen
OpenWrite	FileStream	öffnet eine Datei zum Schreiben
ReadAllBytes	byte[]	öffnet Binärdatei und liest Inhalt in Byte-Array
ReadAllLines	string[]	öffnet Textdatei und liest alle Zueilen in String-Array
ReadAllText	string	öffnet Textdatei. liest alle Zeilen in einen String ein und schliesst Datei
SetAttributes	void	setzt die Dateiattribute

# Methoden der File-Klasse

## Fortsetzung

Die folgende Tabelle gibt eine Übersicht über wichtige statische Methoden:

Methode	Typ	Beschreibung
SetCreationTime SetLastAccessTime SetLastWriteTime	void	setzt Zeitpunkt der Erstellung, des letzten Zugriffs, des letzten Schreibzugriffs
WriteAllBytes	void	erzeugt neue Datei und schreibt Byte-Array hinein
WriteAllLines	void	erzeugt neue Datei und schreibt String-Array hinein
WriteAllText	void	erzeugt neue Datei und schreibt String hinein

# Methoden eines FileInfo-Objekts

Viele Methoden des FileInfo-Objekts haben gleichnamige statische Pendanten in der File-Klasse

Methode	Typ	Beschreibung
AppendText	StreamWriter	fügt Text einer vorhandenen Datei hinzu
CopyTo	FileInfo	kopiert die Datei in ein anderes Verzeichnis
Create	FileStream	erzeugt eine Datei
CreateText	StreamWriter	erzeugt eine Textdatei
Delete	void	löscht eine Datei
Exists	Boolean	prüft das Vorhandensein einer Datei
MoveTo	void	verschiebt die Datei oder benennt sie um
Open	FileStream	öffnet eine Datei
OpenRead	FileStream	öffnet eine Datei für den Lesezugriff
OpenText	StreamReader	öffnet eine Textdatei für den Lesezugriff
OpenWrite	FileStream	öffnet eine Datei für den Schreibzugriff

# Eigenschaften eines FileInfo-Objekts

Eigenschaft	Beschreibung
Attributes	liest oder schreibt die Dateiattribute (ReadOnly, Hidden, usw.)
CreationTime	liest oder schreibt Zeitpunkt der Erstellung einer Datei
Directory	liefert Instanz des übergeordneten Verzeichnisses
DirectoryName	liefert den vollständigen Dateipfad
Extension	liefert Dateierweiterung (z.B. .txt für Textdatei)
FullName	liefert den vollständigen Dateipfad plus Dateinamen
LastAccessTime	liest oder schreibt den Zeitpunkt des letzten Zugriffs
LastWriteTime	liest oder schreibt den Zeitpunkt des letzten Schreibzugriffs
Length	liefert die Dateigröße in Bytes
Name	liefert die vollständigen Dateinamen

# Übungen

## Uebungsaufgaben

"Uebungen\Kurseinheit 7\Übung\Filehandling.pdf"



# Zusätzliche Dokus

## Wichtige String Funktionen

s. `"..\BVS\Repo\BVS\Sonstiges\Wichtige String Funktionen.pdf"`