

Input – Output Műveletek

Tóth Zsolt

Miskolci Egyetem

2013

- 1 I/O Osztályok
 - File-rendszer megvalósítás
 - File-rendszer módosítás, figyelés
 - Stream kezelés
 - IOException és társai
- 2 Alcsomagok
- 3 Példa
 - Property fájl

- Háttértár Információk
- getDrives()
- IsReady()
- Name()
- RootDirectory()
- TotalSize()

DriveType()

- CDRom
- Fixed
- Unknown
- Network
- Removeable

System.IO.DirectoryInfo

- DirectoryInfo-t reprezentál
- Példány szintű metódusok
 - Create()
 - Delete()
 - GetDirectories()
 - GetFiles()
 - GetFiles(String)
 - MoveTo()
- CreationTime
- Exists
- Name
- Parent
- Root

- DirectoryInfo statikus változata
- statikus metódusok gyűjteménye
- CreateDirectory(String)
- Delete(String)
- GetDirectories(String)
- GetFiles(String)

- File reprezentáció
- Példány szintű metódusok
- CopyTo(String)
- Create()
- Delete()
- Open(FileMode)
- CreationTime
- Directory
- Exists
- FullName
- Length
- Name

- FileInfo statikus változata
- statikus metódusok gyűjteménye
- AppendText
- Create(String)
- Delete()
- Exists()

- String alapú
- Elérési út alapú
- Platform "független" műveletek
- Statikus metódusok gyűjteménye
- GetDirectoryName()
- GetFileName()
- GetFullPath()
- GetPathRoot()

- 1 I/O Osztályok
 - File-rendszer megvalósítás
 - File-rendszer módosítás, figyelés
 - Stream kezelés
 - IOException és társai
- 2 Alcsomagok
- 3 Példa
 - Property fájl

Létrehozás, Áthelyezés, Törlés

Create

- Directory.
CreateDirectory()
- DirectoryInfo.
Create()
- File.Create()

Copy, Move

- Directory.Move()
- DirectoryInfo.
MoveTo()
- File.Copy()

Delete

- Directory.Delete()
- DirectoryInfo.
Delete()
- File.Delete()

- Változások figyelése
- Esemény kezelés

Properties

- Filter
- IncludeSubdirectories
- Path

Events

- Changed
- Created
- Deleted
- Error
- Renamed

- 1 I/O Osztályok
 - File-rendszer megvalósítás
 - File-rendszer módosítás, figyelés
 - **Stream kezelés**
 - IOException és társai
- 2 Alcsomagok
- 3 Példa
 - Property fájl

- abstract
- Byte folyam
- Állományok általános kezelése
- abstract
 - ▶ BufferedStream
 - ▶ FileStream
 - ▶ MemoryStream
 - ▶ PrintingQueueStream
 - ▶ CryptoStream
 - ▶ ...

Fő metódusok

- Close()
- Flush()
- Read()
- Seek()
- Write()

FileAccess, FileMode

FileAccess

- Read
- Write
- ReadWrite

FileMode

- CreateNew (IOException)
- Create (Overwrite)
- Open
(FileNotFoundException)
- OpenOrCreate
- Truncate
- Append

System.IO.FileReader

```
//Alapból a <project>/bin/debug mappában futtatja a  
    PlainText.txt pedig a <project> mappában van.  
FileInfo inputFile = new  
    FileInfo("../..../PlainText.txt");  
FileStream fs = new FileStream(  
inputFile.FullName,  
    FileMode.Open,  
    FileAccess.Read);
```

System.IO.FileReader

```
byte[] buffer = new byte[1024];
int end = 0;
while ( (end = fs.Read(buffer, 0, buffer.Length)) >
    0)
{
    for (int i = 0; i < end; i ++)
    {
        Console.Write(buffer[i]);
    }
}
```


System.IO.StreamReader

```
var sr = new
    StreamReader(tempFile.FullName);
string line;
while ((line =
    sr.ReadLine()) !=
    null)
{
    Console.WriteLine(line);
}
sr.Close();
```

- StreamReader(string path)
- ReadLine()
- Close()

System.IO.StreamWriter

- StreamWriter(string path, bool append)
- WriteLine(string line)
- Flush()
- Close()

```
var sw =  
    new StreamWriter(  
        tempFile.FullName,  
        true);  
sw.WriteLine(DateTime.Now);  
sw.Close();
```

- 1 I/O Osztályok
 - File-rendszer megvalósítás
 - File-rendszer módosítás, figyelés
 - Stream kezelés
 - IOException és társai
- 2 Alcsomagok
- 3 Példa
 - Property fájl

- Általános
- Ősosztály
- Minden ami
 - ▶ stream
 - ▶ file
 - ▶ directory
- Utolsó `catch` blokk

System.IO.IOException

- System.IO.DirectoryNotFoundException
- System.IO.DriveNotFoundException
- System.IO.EndOfStreamException
- System.IO.FileLoadException
- System.IO.FileNotFoundException
- System.IO.PathTooLongException
- System.IO.PipeException

{Drive,Directory,File}NotFoundException

- Gyakori
- Fájl vagy Mappa nem található
- Számos metódus dobhatja

Fontosabb kapcsolódó osztályok

- Directory
- DirectoryInfo
- File
- FileInfo
- Path

UnauthorizedAccessException

```
public class UnauthorizedAccessException :  
    SystemException
```

- OS megtagadja a hozzáférést
- IO Error
- Biztonsági hiba
- Számos forrás
 - ▶ FileInfo.Delete()
 - ▶ FileInfo.CreateText()
 - ▶ File.OpenRead()

SecurityException

```
public class SecurityException : SystemException
```

- Biztonsági hiba
- Security Error
- Properties
 - ▶ Action
 - ▶ Method
 - ▶ PermissionType

Kiváltó metódusok

- File.ReadLines()
- File.ReadAllText()
- File.WriteAllText()

Tartalom jegyzék

- 1 I/O Osztályok
 - File-rendszer megvalósítás
 - File-rendszer módosítás, figyelés
 - Stream kezelés
 - IOException és társai

- 2 Alcsomagok

- 3 Példa
 - Property fájl

- Osztály gyűjtemény
- Stream-ek
ki-/becsomagolása
- Zip formátum
 - ▶ GZipStream
 - ▶ ZipFile
 - ▶ ZipArchive
 - ▶ ZipArchiveEntry

Enumerációk

- CompressionLevel
- CompressionMode
- ZipArchiveMode

Assembly: System.IO.Log (in System.IO.Log.dll)

- Naplózási funkciók
- Rekord orientált
- LogStore
- LogExtent
- FileRecordSequence
- LogRecord

- Soros portok kezelése
- Szinkron I/O elérés
- esemény vezérelt I/O elérés
- Stream wrapelés

- 1 I/O Osztályok
 - File-rendszer megvalósítás
 - File-rendszer módosítás, figyelés
 - Stream kezelés
 - IOException és társai
- 2 Alcsomagok
- 3 Példa
 - Property fájl

- .properties
- ASCII szöveges állomány
- Alkalmazások
 - ▶ Alkalmazás konfiguráció
 - ▶ Honosítás
 - ▶ Nemzetköziesítés
- Java világra jellemző

Többféle formátum

- key=value
- key = value
- key:value
- key value
- #comment

Property fájl

```
Dictionary<string, string> properties = new
    Dictionary<string, string>();
foreach (var row in
    File.ReadAllLines("../../lotto.properties"))
{
    //String feldolgozas
    //soronkent
}
```