

Trabalhando com Arquivos e Streams

Ricardo Augusto Vicentini
Senior Software Engineer - Nubank

Mais sobre mim

- Desenvolvedor desde 2002
- Entrei na área de desenvolvimento porque sempre gostei muito de ***Games***.
- Entusiasta da linguagem C#, escovador de bit 🤪
- Tenho Gatos, Cachorros e curto d+ Games e Aviões.



Objetivo do curso

- Aprender como manipular arquivos e pastas;
- Como navegar pela estrutura de diretórios do SO;
- Modificar arquivos existentes;
- Criar novos arquivos em memória e salvá-los;
- Diferentes técnicas de importação de dados;
- Criar, Validar e consumir arquivos CSV;

Percurso

Aula 1

Namespace System.IO e suas principais classes

Aula 2

Capturando eventos de mudança em arquivos e diretórios

Aula 3

Lendo e escrevendo arquivos com Streams

Aula 4

Importação de dados (alimentando sistemas)

Aula 1: System.IO

Arquivos e streams

Namespace que agrega um conjunto de classes, estruturas, enumeradores e delegates, relativos a operações de Entrada e Saída de dados (Input/Output)



Principais classes

- **File e FileInfo**

fornece suporte para criar, copiar apagar, mover e abrir arquivos de forma individual

- **Directory e DirectoryInfo**

fornece suporte para criar, mover e percorrer diretórios e subdiretórios

- **FileSystemWatcher**

monitora mudança no sistema de arquivos, através de disparo de eventos quando um arquivo ou diretório muda.

- **FileStream, MemoryStream, StreamReader**

Classe File e FileInfo

Podemos realizar operações comuns:

- Criar
- Abir
- Deletar
- Copiar
- Mover
- renomear

Classe File e FileInfo

Podemos alterar permissões, nível de acesso e modo de criação/abertura, através dos enumeradores:

- FileAccess
- FileShare
- FileMode



Classe Directory e DirectoryInfo

Semelhante a classe File, podemos realizar operações tais como, criar, mover, renomear, excluir e enumerar diretórios e subdiretórios



Classe DirectoryInfo

Propriedades

- Exists
Boolean que indica se o diretório existe.
- Name
Nome do diretório, e para obter o caminho “Path” FullName
- Parent
Retorna o diretório Pai, pode retornar null se estivermos em uma pasta raiz “root”
- Root
Retorna uma instância de DirectoryInfo que representa o diretório raiz



Vamos criar nosso primeiro arquivo de texto usando a classe estática File

**Escrever texto no arquivo utilizando o método
CreateText**

**Solicitar do usuário o nome do arquivo e tratar possíveis
erros**

Verificar se o arquivo já existe

Utilizando a classe estática Directory vamos aprender a:

- **Criar diretórios e subdiretórios**
- **Copiar e mover arquivos de um diretório para outro**
- **Apagar diretórios**
- **Navegar entre pastas, localizar pastas e arquivos**



DIGITAL
INNOVATION
ONE

Aula 2: FileSystemWatcher

Arquivos e streams

Utilização

Permite monitorar eventos de modificação em diretórios e arquivos, através do disparo de eventos.

Eventos

- OnChanged
- OnCreated
- OnDeleted
- OnRenamed
- OnError



Podemos indicar nesse filtro a quais arquivos ou extensões queremos monitorar

- **Criar uma console app e monitorar mudanças em uma pasta**



DIGITAL
INNOVATION
ONE

Aula 3: Lendo e escrevendo arquivos com Streams

Arquivos e streams

Definição

Stream é uma classe abstrata que fornece uma visão genérica de uma sequência de bytes.

Existe uma grande vantagem de se utilizar Streams, pois os dados são escritos, lidos ou modificados através de um ***Buffer***.

Buffer

Espécie de reservatório onde podemos armazenar uma grande quantidade de dados em pequenos pedaços. O famoso dividir para conquistar.



DIGITAL
INNOVATION
ONE

Buffer



Buffer - files



Buffer - files





DIGITAL
INNOVATION
ONE

Buffer



Demos

Utilizar StringWriter, StringReader, StreamWriter, StreamReader

Aula 4: Importar dados

Arquivos e Streams

Necessidade

Sistemas precisam se comunicar e isso pode ocorrer de diversas formas, como por exemplo: APIs Rest, Mensageria e até mesmo troca de arquivo

Formatos

O formato mais utilizado na atualidade é o Json

Java Script Object Notation

Mas temos outros como:

XML, Positional, Tab Delimited e CSV



Comma-separated Values

- Os dados são separados por vírgula;
- Podem ou não conter aspas para ajudar na identificação do formato contido em cada coluna;
- Podem ou não possuir cabeçalho;
- Podem utilizar outro tipo de delimitador;



Demos

Escrever arquivos CSV utilizando StreamWriter

Ler arquivos CSV utilizando StreamReader

Tratamento a falhas

Lidar com arquivos grandes

biblioteca csvHelper



DIGITAL
INNOVATION
ONE



Level UP!





Review

- **File, Directory, FileInfo e DirectoryInfo**
- **FileSystemWatcher**
- **StringWriter, StringReader, StreamWriter e StreamReader**
- **Buffer**
- **Como importar e exportar dados utilizando arquivos CSV**

Documentação

<https://docs.microsoft.com/pt-br/dotnet/api/system.io?view=net-6.0>

<https://joshclose.github.io/CsvHelper/>

<https://www.nuget.org/>

Github - Demos

https://github.com/ricardovicentini/arquivos_e_streams/tree/master