Reading and Writing CSV Data

Jason Roberts

.NET DEVELOPER

@robertsjason   dontcodetired.com

Overview

An overview of the CsvHelper library

Reading CSV data dynamically

Trimming spaces

Allowing CSV comments

Ignoring blank lines

Configuring a custom delimiter

Configuring CSV file headers

Reading data into entity classes

Configuring custom class mapping

Creating custom type convertors

Writing CSV files

An Overview of the CsvHelper Library

https://github.com/JoshClose/CsvHelper

https://www.nuget.org/packages/CsvHelper

45 million downloads

Read & write CSV data

Design goals:
- Flexibility
- Performance
- Ease of use

Defaults:
- Conservative writing
- Liberal reading

Highly configurable

Summary

An overview of the CsvHelper library

csvReader.GetRecords<dynamic>();

CsvReader configuration:
- CsvConfiguration.TrimOptions
- CsvConfiguration.AllowComments
- CsvConfiguration.IgnoreBlankLines
- CsvConfiguration.Delimiter
- CsvConfiguration.HasHeaderRecord

csvReader.GetRecords<Order>()

ClassMap<ProcessedOrder>

ITypeConverter

csvWriter.WriteRecords(records)

Up Next:

Creating Automated Tests for File Access Code