CSV Reader/Writer

Generated by Doxygen 1.9.1

Chapter 1

A very tiny simple CSV file reader

Not for a professional use, but for just visualize, manipulate and print your comma separated data.

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

Class Documentation

3.1 CSVFile< T > Class Template Reference

`#include <CSVFile.h>`

Public Member Functions

- **CSVFile** (std::vector< std::string > header, T **data, int rows, int cols)
- void **head** (int heads=5)
- std::vector< std::string > **getHeader** ()
- std::vector< std::vector< T > > **getData** ()
- void **appendToHeader** (std::string element)
- void **appendRowToData** (std::vector< T > row)
- int **getDataRows** ()
- int **getDataCols** ()

3.1.1 Detailed Description

template<class T>
class CSVFile< T >

This is CSVFile class, use this calss for manage or create your CSV file. This class has two attributes:

- `header` (`vector<string>`) : Usually a CSV File contains an header for Columns Names
- data (`vector<vector<T>>`): The data (numerical) contained in the file.

Constructor:

- No params: Creates empty CSVFile Object

Constructor with Parameters:

- `vector<string>` header: if you want to build CSVFile object starting from aleady existing data

- `T**` data : if you have a matrix of numerical data

Methods:

- appendToHeader: Append element to header (kept private)
- appendRowToData: Append a vector of values to data matrix (kept private)
- getters

The documentation for this class was generated from the following file:

- CSVFile.h

3.2 CSVRW< T > Class Template Reference

`#include <CSVRW.h>`

Public Member Functions

- **CSVRW** (CSVRW &other)=delete
- void **operator=** (const CSVRW &)=delete
- void **read_file** (std::string filepath, CSVFile< T > *file, bool header=true, char delim=',')
- void **write_file** (std::string filename, CSVFile< T > *file, char delim=',')

Static Public Member Functions

- static CSVRW * **instance** (dtypes dt)

3.2.1 Detailed Description

template<class T>
class CSVRW< T >

This is the CSVRW (CSV Read & Write) This class can help to read and write CSV files just instanciating one single time. A Singleton design Pattern is being used to avoid multiple reader/writer instancing.

A global enum variable is needed for manual data type conversion (not more needed with C++20):

- `F` for float
- `D` for double
- `I` for integer

So this class needs to be instanciated by using this dtype flags. Manual data type control can improve memory space allocation.

Class Methods:

- `read_file`: method for reading a csv file by provinding local path, user can provide custom delimiter
- `write_file`: method for writing data on a csv file, user can provide custom delimiter

Function read_file parameters:

- `filepath`: path/to/file (string)
- `CSVFile<T> *file`: CSVFile variable to store fetched data
- `header`: is your file having an header? true/false (default true)
- `delim`: delimiter character (default ',')

Function write_file parameters:

- `filename`: `path/to/write/` (must .csv extension be provided)
- `file`: pointer to `CSVFile` variable (can be created starting from numerical matrix)
- `delim`: delimiter (default ',')

The documentation for this class was generated from the following file:

- CSVRW.h

Index