



The CSVStreamClass

◆ What is a CSV file?



The CSVStreamClass

- ◆ What is a CSV file?
 - By convention
 - ◆ **Fields are normally delimited by commas**



The CSVStreamClass

◆ What is a CSV file?

- By convention

- ◆ Fields are normally delimited by commas
- ◆ **Literals are qualified by double-quotes**



The CSVStreamClass

◆ What is a CSV file?

■ By convention

- ◆ Fields are normally delimited by commas
- ◆ Literals are qualified by double-quotes
- ◆ **Commas and new line characters may appear within qualified literals**



The CSVStreamClass

◆ What is a CSV file?

■ By convention

- ◆ Fields are normally delimited by commas
- ◆ Literals are qualified by double-quotes
- ◆ Commas and new line characters may appear within qualified literals
- ◆ **Qualifiers may appear self-escaped within literal strings**



The CSVStreamClass

◆ What is a CSV file?

■ By convention

- ◆ Fields are normally delimited by commas
- ◆ Literals are qualified by double-quotes
- ◆ Commas and new line characters may appear within qualified literals
- ◆ Qualifiers may appear self-escaped within literal strings
- ◆ **Sometimes the first row of data is actually a header containing field names**



The CSVStreamClass

◆ History

- **Discussion on comp.lang.rexx**



The CSVStreamClass

◆ History

- Discussion on `comp.lang.rexx`
- **Classic REXX sub-routine**



The CSVStreamClass

◆ History

- Discussion on comp.lang.rexx
- Classic Rexx sub-routine
- **Producing the class for ooRexx**



The CSVStreamClass

◆ History

- Discussion on comp.lang.rexx
- Classic REXX sub-routine
- Producing the class for ooRexx
- **Included in the 4.1.0 release**
 - ◆ As an extension
 - `::requires "csvstream.cls"`



The CSVStreamClass

◆ Subclasses the stream class



The CSVStreamClass

◆ Subclasses the stream class

◆ **New Methods**

- CSVLineIn
- CSVLineout
- GetHeaders
- SetHeaders



The CSVStreamClass

- ◆ Subclasses the stream class
- ◆ **Overrides methods**
 - Init
 - Open
 - Close



The CSVStreamClass

◆ Subclasses the stream class

◆ **New Attributes**

- Delimiter
- Qualifier
- StripOption
- StripChar
- SkipHeaders





The CSVStreamClass

◆ Subclasses the stream class

◆ **New Attributes**

- Headers~Field(n)~Name
- Headers~Field(n)~Literal

- RawData (after a csvLineIn)
- Values (a table)



Using the CSVstream Class

- ◆ Files without a header row
 - Writing to the file
 - ◆ Data to be written may come from
 - An array
 - A queue (or a circularQueue)
 - A list
 - ◆ Data read is returned in an array



Using the CSVstream Class

```
csv = .csvStream~new('c:\test.csv')
csv~open('write replace')  -- Always open csvStreams explicitly
csv~csvLineout(.array~of('Line 1 Field 1', 'Line 1 Field 2', 'Line 1 Field 3'))
csv~csvLineout(.array~of('Line 2 Field 1', 'Line 2 Field 2', 'Line 2 Field 3'))
csv~close
```

```
csv~open('read')
line = 0
do while csv~chars > 0
  line += 1
  csvArray = csv~csvLineIn
  do field = 1 to csvArray~items
    say 'line' line 'Field' field ':' csvArray[field]
  end /* DO */
end /* DO */
csv~close
```

```
::requires 'CSVStream.cls'
```

file



Using the CSVstream Class

- ◆ Files with a header row
- ◆ Indicate with an 'h' on initiation
 - Writing to the file
 - ◆ Data to be written may come from
 - An array
 - A table
 - A stem
 - ◆ Data read is returned in an array & also available as a table (the attribute values)



Using the CSVstream Class

```
csv = .csvStream~new('c:\headered.csv','h')
csv~open                                -- csvStreams must be opened explicitly

do while csv~chars > 0
  array = csv~csvLineIn
  say
  say 'line =' array~makeString('l',',')
  do field over csv~values              -- cannot rely on the order of fields
    say field ':' csv~values~at(field)
  end /* DO */
end /* DO */

csv~close

::requires 'CSVStream.cls'
```



Using the CSVstream Class

Source file:

```
"Forename","Surname"  
"Jon","Wolfers"  
"Rene","Jansen"  
"Rony","Flatscher"
```

Results:

```
line = Jon,Wolfers  
Surname: Wolfers  
Forename: Jon
```

```
line = Rene,Jansen  
Surname: Jansen  
Forename: Rene
```

```
line = Rony,Flatscher  
Surname: Flatscher  
Forename: Rony
```



Using the CSVstream Class

◆ Other Features

■ Delimiter attribute

```
csv = .csvStream~new('c:\continental.csv', 'h')  
csv~delimiter = ';'   
csv~open
```



Using the CSVstream Class

◆ Other Features

■ Delimiter attribute

```
csv = .csvStream~new('c:\continental.csv', 'h')  
csv~delimiter = ';'   
csv~open
```

■ **StripOption and StripChar attributes**

- ◆ Allow one to strip the data as you read the file



The CSVStreamClass

Questions?