csv file import, export

kacper.topolnicki@uj.edu.pl

The `numpycsv.py` script contains a demonstration of of `csv` file creation and export. These types of files can be read by spreadsheet programs such as *MS Exel* or *LibreOffice Calc*.

Comments

The script contains many comments. Any lines that begin with the `#` symbol are ignored by python and only contain additional information for the programmer. The script file contains many instances of `#@`, `#@ref`, . . . These lines are used by an external programm to create a PDF file, these lines can also be ignored by the programmer.

Running the script

To run the script simply navigate to this directory in the terminal and run:

```
<user> $ python numpycsv.py
```

Alternatively you can make the script executable and run:

```
<user> $ ./numpycsv.py
```

You can also run `ipython` and execute the commands one by one.

Importing the necessary libraries

First we import the `numpy` library [numpycsv.py line: 48]

```
import numpy
```

Creating a `csv` file

Next we create a sample `numpy` array using the `asarray` function [numpycsv.py line: 63]

```
array = numpy.asarray([[1 , 2 ,3],[4 , 5 , 6],[7 , 8 , 9]])
```

and export this data to `array.csv` [numpycsv.py line: 67]

```
numpy.savetxt("array.csv" , array , delimiter = ',')
```

You can try opening this file in a spreadsheet application.

Importing a csv file

The file we just created can be imported, converted to a `numpy` array and stored in a variable [numpycsv.py line: 79]

```
readarray = numpy.genfromtxt("array.csv" , delimiter = ",")
```

We can print the matrix to check if importing succeeded [numpycsv.py line: 92]

```
print(readarray)
```

Finally, we can cast the elements of the imported array to the integer type (originally, elements of `array` were integers). [numpycsv.py line: 96]

```
print(readarray.astype("int"))
```