# Story 1: Display CSV as Table

Write an application to view CSV files on the console. [1]It ́s supposed to be a very simple program: just call it from the command line like this

C:\>csvviewer.exe persons.csv

…and it will display the file content in a table like that: [2]

Name|Age|City |

-----+---+--------+

Peter|42 |New York|

Paul |57 |London |

Mary |35 |Munich |

The data displayed as a table with a header line and cell borders like depicted. The corresponding file looks like follows:

Name;Age;City

Peter;42;NewYork

Paul;57;London

Mary;35;Munich

Each line contains a data record; the first line contains the column names; columns are separated by „;“. The data record fields’ content is simple: no line breaks, no field separators, no need to process values put in „“. The encoding of the CSV text files is UTF-8.

The table should have fixed width columns matching the longest value for each column.

--------------------------------------------------------------------------------

[1] A console frontend is used for this exercise because that is the least common denominator in terms of frontend for all platforms. Whether you ́re doing Ruby or C# or Java you should be able to put a console frontend on top of any business logic. Sure a GUI would be more stylish, but it also would distract from the main challenges of the exercise.

[2] Assume all CSV files to contain only lines of a length fitting a display line; no horizontal scrolling needed. Very likely all lines are of different length, though.

# Story 2: Pagination

The viewer will display the file ́s content in a page wise manner like that:

Name|Age|City |

-----+---+--------+

Peter|42 |New York|

Paul |57 |London |

Mary |35 |Munich |

N(ext page, P(revious page, F(irst page, L(ast page, eX(it

Pages of data records are displayed as a table with a header line and cell borders like depicted. The corresponding file looks like follows; when assuming a page length of 3 data records it would be displayed as three pages:

Name;Age;City

Peter;42;NewYork

Paul;57;London

Mary;35;Munich

Jaques;66;Paris

Yuri;23;Moscow

Stephanie;47;Stockholm

Nadia;29;Madrid

The page length should have a reasonable default for your platforms console windows, but it also can be set by passing it to the application like this:

C:\>csvviewer.exe persons.csv 40

The page table should have fixed width columns matching the longest value for each column within a page.

# Story 3: Advanced Pagination

1.Each data record should be preceded by a record number.

2.Show which page is displayed out of how many in total below the table.

3.Let the user jump to a page by entering its number.

The display should look like this:

No.|Name |Age|City |

---+-----+---+--------+

1.|Peter|42 |New York|

2.|Paul |57 |London |

3.|Mary |35 |Munich |

Page 1 of 3

N(ext page, P(revious page, F(irst page, L(ast page, J(ump to page, eX(it

# Story 4: Persistence

Enhance the CSV viewer so that when opening the application again on a CSV file it has already processed (partly or fully)[1], processing does not start all over. Instead processing should continue from what position in the CSV file it had reached last time when the application was exited.

--------------------------------------------------------------------------------

[1] …and the CSV file has not changed since then

# Story 5: Huge Files

Make the CSV viewer capable of displaying huge CSV files; think 3 times the GB of your computer’s memory. Yes, the CSV viewer should be able to handle billions of data records, or at least many more records than would fit into memory.

Still, when the application is started, the first page should be displayed immediately. But of course then the maximum number of pages cannot be known – yet. As long[1] as that´s the case, just put a “?” after it:

No.|Name |Age|City |

---+-----+---+--------+

1. |Peter|42 |New York|

2. |Paul |57 |London |

3. |Mary |35 |Munich |

Page 1 of 3?

N(ext page, P(revious page, F(irst page, L(ast page, J(ump to page, eX(it

--------------------------------------------------------------------------------

[1] This implies there´s background processing taking place. The file has to be further processed after displaying the first page, because the user wants to know the exact number of pages and wants to be able to immediately jump to any page (that has already been processed). With regard to high usability it´s no option to only start reading from the file when the user has entered a page number to jump to.