300ump.csv

The data starts on the 3rd line instead of the second. This tests the templates ability to change where the data starts. This file also tests how the program handles scientific notation.

Backslaash.csv

Tests how the program handles the backslash delimeter, must be specified with a template

Basic\_errors.csv

Simple file with a few supported types and some basic errors

* Col1: type – integer
  + Errors – row 21 – float
  + Row 25 - float
* Col2: type – float
  + Errors – row 2- int
  + row 6 – outlier/int
  + row 18 - int
  + Row 19 – outlier
* Col3:type –day
  + Errors – row 16 – typo
* Col4: type – cool
  + Errors: row 7 – typo
* Col5: type – sci notation
  + Errors: row6 – outlier
  + Row 7 - outlier
  + Row 8 – wrong type
  + Row 14 – outlier
  + Row 15 – int
  + Row 20 – float
* Col6: type – indentifier
  + Errors: row 4 – too long
  + Row 8 – too long
  + Row 14 – too short
  + Row 24, 25 – too long

Dash.csv

File separated by dashes, to test automated delimiter detection

oil\_originial.csv

File with large amount of columns, contains wrong formatting and many different data types

Pipe.csv

Tests the program automatic detection of pipe delimited data

Semicolon.csv

Tests the program automated detection of semicolon delimited data

Space.csv

Tests the programs automated detection of space delimited data, have to use template

Tabs.csv

Tests the programs automated detection of tab delimited data