process_clean_files

May 14, 2019

1 Clean, process dates in text files

Material for the hackathon ENSAE / Red Cross / 2015. Cleaning the data, change encoding.

```
In [1]: %matplotlib inline
        import matplotlib.pyplot as plt
        plt.style.use('ggplot')
```

```
In [2]: from jyquickhelper import add_notebook_menu
        add_notebook_menu()
```

```
Out[2]: <IPython.core.display.HTML object>
```

1.1 prepare data

```
In [3]: import ensae_projects
```

```
In [4]: %ls
```

```
 Volume in drive C has no label.
 Volume Serial Number is F074-BCDF

 Directory of C:\PythonENSAE11\workspace

11/25/2015  03:11 PM    <DIR>          .
11/25/2015  03:11 PM    <DIR>          ..
11/25/2015  03:06 PM    <DIR>          .ipynb_checkpoints
11/22/2015  05:23 PM    <DIR>          docs
11/25/2015  02:23 PM            16,704 download_data_azure.ipynb
11/25/2015  02:10 PM             5,658 ITMMASTER.schema.txt
11/25/2015  02:22 PM       103,096,479 ITMMASTER.txt
11/25/2015  03:11 PM             8,463 process_file.ipynb
11/25/2015  03:11 PM        51,798,216 SINVOICE_.clean.txt
11/25/2015  02:11 PM     1,362,433,753 SINVOICE_.txt
11/25/2015  02:13 PM     1,252,461,865 SINVOICEV_.txt
11/25/2015  02:21 PM     8,821,375,868 stojou.csv
               8 File(s) 11,591,197,006 bytes
               4 Dir(s)  48,990,134,272 bytes free
```

```
In [5]: from ensae_projects.datainc import change_encoding, convert_dates
        from pyquickhelper.loghelper import fLOG
        fLOG(OutputPrint=True)
        fLOG("start")
```

```
2015-11-25 15:13:11 start
```

```
In [6]: def transform(i, line):
            spl = line.split("\t")
            if i == 0:
                spl = [_.replace("_0", "").strip('" ') for _ in spl]
            else:
                spl = [convert_dates(_, 'F').strip('" ') for _ in spl]
            return "\t".join(spl)

        change_encoding("SINVOICE_.txt", "SINVOICE_.clean.txt", "latin-1", "latin-1", process=transform
```

```
2015-11-25 15:13:24 SINVOICE_.txt - 10000 lines
2015-11-25 15:13:36 SINVOICE_.txt - 20000 lines
2015-11-25 15:13:49 SINVOICE_.txt - 30000 lines
2015-11-25 15:14:02 SINVOICE_.txt - 40000 lines
2015-11-25 15:14:15 SINVOICE_.txt - 50000 lines
2015-11-25 15:14:28 SINVOICE_.txt - 60000 lines
2015-11-25 15:14:41 SINVOICE_.txt - 70000 lines
2015-11-25 15:14:53 SINVOICE_.txt - 80000 lines
2015-11-25 15:15:06 SINVOICE_.txt - 90000 lines
2015-11-25 15:15:19 SINVOICE_.txt - 100000 lines
2015-11-25 15:15:32 SINVOICE_.txt - 110000 lines
2015-11-25 15:15:45 SINVOICE_.txt - 120000 lines
2015-11-25 15:15:57 SINVOICE_.txt - 130000 lines
2015-11-25 15:16:10 SINVOICE_.txt - 140000 lines
2015-11-25 15:16:23 SINVOICE_.txt - 150000 lines
2015-11-25 15:16:36 SINVOICE_.txt - 160000 lines
2015-11-25 15:16:49 SINVOICE_.txt - 170000 lines
2015-11-25 15:17:02 SINVOICE_.txt - 180000 lines
2015-11-25 15:17:15 SINVOICE_.txt - 190000 lines
2015-11-25 15:17:27 SINVOICE_.txt - 200000 lines
2015-11-25 15:17:40 SINVOICE_.txt - 210000 lines
2015-11-25 15:17:53 SINVOICE_.txt - 220000 lines
2015-11-25 15:18:06 SINVOICE_.txt - 230000 lines
2015-11-25 15:18:19 SINVOICE_.txt - 240000 lines
2015-11-25 15:18:32 SINVOICE_.txt - 250000 lines
2015-11-25 15:18:45 SINVOICE_.txt - 260000 lines
2015-11-25 15:18:58 SINVOICE_.txt - 270000 lines
2015-11-25 15:19:11 SINVOICE_.txt - 280000 lines
2015-11-25 15:19:24 SINVOICE_.txt - 290000 lines
2015-11-25 15:19:37 SINVOICE_.txt - 300000 lines
2015-11-25 15:19:49 SINVOICE_.txt - 310000 lines
2015-11-25 15:20:03 SINVOICE_.txt - 320000 lines
2015-11-25 15:20:16 SINVOICE_.txt - 330000 lines
2015-11-25 15:20:29 SINVOICE_.txt - 340000 lines
2015-11-25 15:20:43 SINVOICE_.txt - 350000 lines
2015-11-25 15:20:57 SINVOICE_.txt - 360000 lines
2015-11-25 15:21:10 SINVOICE_.txt - 370000 lines
2015-11-25 15:21:23 SINVOICE_.txt - 380000 lines
2015-11-25 15:21:36 SINVOICE_.txt - 390000 lines
2015-11-25 15:21:49 SINVOICE_.txt - 400000 lines
```

```
In [7]: change_encoding("SINVOICEV_.txt", "SINVOICEV_.clean.txt", "latin-1", "latin-1", process=transfo
```

```
In [8]: def transform2(i, line):
            spl = line.split(",")
            spl = [convert_dates(_, 'F').strip('" ') for _ in spl]
            return "\t".join(spl)

        change_encoding("ITMMASTER.txt", "ITMMASTER.clean.txt", "latin-1", "latin-1", process=transform

In [9]: def transform3(i, line):
            spl = line.split(",")
            if i == 0:
                spl = [_.replace("_0", "").strip('" ') for _ in spl]
            else:
                spl = [convert_dates(_, 'F').strip('" ') for _ in spl]
            return "\t".join(spl)

        change_encoding("stojou.csv", "stojou.clean.txt", "latin-1", "latin-1", process=transform3, fLO
```

Le dernier exemple est plus lent mais il permet de traiter le où les lignes dont les valeurs incluent le séparateur de colonnes.

```
In [10]: from ensae_projects.datainc import change_encoding_improve, convert_dates
         from pyquickhelper.loghelper import fLOG
         fLOG(OutputPrint=True)
         import re

         reg = re.compile(';"(.*?)"')

         def transform4(i, line, hist):
             a, b = clean_column_name_sql_dump(i, line, hist)
             return a.replace("_0", ""), b

         change_encoding_improve("export_SINVOICE.csv", "export_SINVOICE.clean2.txt", "latin-1", "latin
                                 process=transform4, fLOG=fLOG)
```