

November 6, 2022

```
[2]: %pip install ngutils
```

```
Collecting ngutils
  Downloading ngutils-0.10.1-py3-none-any.whl (19 kB)
Requirement already satisfied: pandas>=1.1.0 in /opt/conda/lib/python3.10/site-packages (from ngutils) (1.5.0)
Requirement already satisfied: tqdm>=4.50.0 in /opt/conda/lib/python3.10/site-packages (from ngutils) (4.64.1)
Requirement already satisfied: lxml>=4.5.2 in /opt/conda/lib/python3.10/site-packages (from ngutils) (4.9.1)
Requirement already satisfied: requests>=2.24.0 in /opt/conda/lib/python3.10/site-packages (from ngutils) (2.28.1)
Requirement already satisfied: python-dateutil>=2.8.1 in /opt/conda/lib/python3.10/site-packages (from pandas>=1.1.0->ngutils) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /opt/conda/lib/python3.10/site-packages (from pandas>=1.1.0->ngutils) (2022.2.1)
Requirement already satisfied: numpy>=1.21.0 in /opt/conda/lib/python3.10/site-packages (from pandas>=1.1.0->ngutils) (1.22.4)
Requirement already satisfied: idna<4,>=2.5 in /opt/conda/lib/python3.10/site-packages (from requests>=2.24.0->ngutils) (3.4)
Requirement already satisfied: charset-normalizer<3,>=2 in /opt/conda/lib/python3.10/site-packages (from requests>=2.24.0->ngutils) (2.1.1)
Requirement already satisfied: certifi>=2017.4.17 in /opt/conda/lib/python3.10/site-packages (from requests>=2.24.0->ngutils) (2022.9.24)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /opt/conda/lib/python3.10/site-packages (from requests>=2.24.0->ngutils) (1.26.11)
Requirement already satisfied: six>=1.5 in /opt/conda/lib/python3.10/site-packages (from python-dateutil>=2.8.1->pandas>=1.1.0->ngutils) (1.16.0)
Installing collected packages: ngutils
Successfully installed ngutils-0.10.1
Note: you may need to restart the kernel to use updated packages.
```

```
[153]: import io, os
import pandas as pd
import numpy as np
import datetime
```

```

from tqdm import tqdm
from ngutils import Tc

import concurrent.futures

import requests
from sklearn.utils import gen_batches

api_key = '6655b49c72afdc5d55edbf58bb4a9b6d'

tc = Tc()
tc.log('start')

```

start: 2022-11-06 18:40 mem:9.371GB

```

[4]: # %%time
      # df = pd.read_pickle('../data/Full_16_09_22.pickle')

```

CPU times: user 24.7 s, sys: 7.18 s, total: 31.9 s  
Wall time: 31.9 s

```

[5]: %%time
      df = pd.read_pickle('../data/Full_16_09_22-cat.pickle')

```

CPU times: user 10.4 s, sys: 3.42 s, total: 13.9 s  
Wall time: 13.9 s

```

[23]: %%time
      #
      df[df.duplicated(subset=['          '], keep=False)].sort_values('          ')

      #          174

```

CPU times: user 4.74 s, sys: 84.4 ms, total: 4.82 s  
Wall time: 4.8 s

```

[23]:
      \
1401      73413460      73417854      00001402/21
1152234      73413460      73417854      00001402/21
1152235      73418979      73438385      00002693/21
2692      73418979      73438385      00002693/21
3394      73422028      73493222      00003395/21
...      ...      ...      ...
538620      76030186      76047704      00539016/21
1152406      76034619      76035077      00539965/21
539569      76034619      76035077      00539965/21
540342      76038412      76174932      00540738/21

```

1152407                      76038412                      76174932    00540738/21

(mos.ru) \

1401	NaN
1152234	NaN
1152235	NaN
2692	NaN
3394	NaN
...	...
538620	NaN
1152406	NaN
539569	NaN
540342	NaN
1152407	NaN

Timezone \

1401	2021-01-01 10:32:00.823530+03:00
1152234	2021-01-01 10:32:00.823530+03:00
1152235	2021-01-01 12:43:24.546609+03:00
2692	2021-01-01 12:43:24.546609+03:00
3394	2021-01-01 13:48:35.566341+03:00
...	...
538620	2021-01-31 15:09:30.019257+03:00
1152406	2021-01-31 16:11:06.318965+03:00
539569	2021-01-31 16:11:06.318965+03:00
540342	2021-01-31 17:07:45.046024+03:00
1152407	2021-01-31 17:07:45.046024+03:00

Timezone \

1401	2021-01-01 12:16:29.483998+03:00
1152234	2021-01-01 12:16:29.483998+03:00
1152235	2021-01-01 19:53:36.113079+03:00
2692	2021-01-01 19:53:36.113079+03:00
3394	2021-01-03 10:31:30.282955+03:00
...	...
538620	2021-01-31 19:45:25.050600+03:00
1152406	2021-01-31 16:18:11.989116+03:00
539569	2021-01-31 16:18:11.989116+03:00
540342	2021-02-02 10:35:48.660509+03:00
1152407	2021-02-02 10:35:48.660509+03:00

\

1401  
1152234  
1152235  
2692  
3394

...  
 538620  
 1152406  
 539569  
 540342  
 1152407

...

1401	phone	frontline
1152234	phone	frontline
1152235	phone	frontline
2692	phone	frontline
3394	phone	frontline

\

...	...	...
538620	phone	frontline
1152406	elec_house	ElecH
539569	elec_house	ElecH
540342	phone	frontline
1152407	phone	frontline

...

\

1401	...	...	2021-01-01	12:16:29.483998+03:00
1152234	...	...	2021-01-01	12:16:29.483998+03:00
1152235	...	...	2021-01-01	19:53:36.113079+03:00
2692	...	...	2021-01-01	19:53:36.113079+03:00
3394	...	...	2021-01-03	10:31:30.282955+03:00
...	...	...	...	...
538620	...	...	2021-01-31	19:45:25.050600+03:00
1152406	...	...	2021-01-31	16:18:11.989116+03:00
539569	...	...	2021-01-31	16:18:11.989116+03:00
540342	...	...	2021-01-31	17:15:44.930874+03:00
1152407	...	...	2021-01-31	17:15:44.930874+03:00

\

1401	NaT	NaT
1152234	NaT	NaT
1152235	NaT	NaT
2692	NaT	NaT
3394	NaT	NaT
...	...	...
538620	NaT	NaT
1152406	2021-02-01 16:39:00+03:00	2021-02-01 17:39:00+03:00
539569	2021-02-01 16:39:00+03:00	2021-02-01 17:39:00+03:00
540342	NaT	NaT
1152407	NaT	NaT

/

\

1401		NaT		NaN
1152234		NaT		NaN
1152235		NaT		NaN
2692		NaT		NaN
3394		NaT		NaN
...		...	...	
538620		NaT		NaN
1152406	2021-04-16 01:38:43.991846+03:00			NaN
539569	2021-04-16 01:38:43.991846+03:00			NaN
540342	2021-04-16 01:38:43.991846+03:00			
1152407	2021-04-16 01:38:43.991846+03:00			

1401		NaN		NaN
1152234		NaN		NaN
1152235		NaN		NaN
2692		NaN		NaN
3394		NaN		NaN
...		...	...	
538620		NaN		NaN
1152406			awful	
539569			awful	
540342				bad
1152407				bad

1401			free
1152234			free
1152235			free
2692			free
3394			free
...		...	...
538620			free
1152406			free
539569			free
540342			free
1152407			free

1401	
1152234	
1152235	
2692	
3394	
...	...
538620	
1152406	

539569  
540342  
1152407

[348 rows x 68 columns]

```
[35]: api_key = api_key[::-1]
```

```
[35]: url = 'https://apidata.mos.ru/v1/datasets/60562/count'
      params = {'api_key': api_key}
      n_rows = int(requests.get(url, params=params).text)
      print(f"                                : {n_rows}")
```

:

481614

```
[ ]: url = 'https://apidata.mos.ru/v1/datasets/60562/rows'
```

```
[ ]: def load_60562(batch):
      params = {'api_key': api_key, '$skip': batch.start, '$top': min(500, batch.
      ↪stop-batch.start)}
      data = ['SIMPLE_ADDRESS', 'UNOM', 'ADM_AREA', 'DISTRICT', 'geoData']
      r = requests.post(url, params=params, json=data)
      with open(f"..data/apidata_mos_ru_60562/{batch.start:06}-{batch.stop-1:06}.
      ↪txt", 'wb') as f:
          f.write(r.content)

      with concurrent.futures.ThreadPoolExecutor(max_workers=16) as executor:
          future_to_file = [executor.submit(load_60562, batch) for batch in
          ↪gen_batches(n_rows, 500)]
          for future in tqdm(concurrent.futures.as_completed(future_to_file),
          ↪total=n_rows//500):
              pass
```

```
[ ]: url = 'https://apidata.mos.ru/v1/datasets/60562/rows'
      params = {'api_key': api_key,
                '$filter': 'Cells/UNOM eq 5783', '$top':2}
      r = requests.get(url, params=params)
      print(r.status_code, r.reason)
      print(f"                                UNOM{r.text}")
```

```
[60]: %%time
```

```
#                UNOM                .                UNOM

import concurrent.futures
import time
```

```

filelist = os.listdir('../data/apidata_mos_ru_60562')
list_df = []
err_fl = []
if os.path.isfile('../data/UNOM_geoData.pickle'):
    set_BATCH = set(pd.read_pickle('../data/UNOM_geoData.pickle')['BATCH'].
↳to_list())
else:
    set_BATCH = set()

def parse_60562(filename):
    try:
        df = pd.read_json(f"../data/apidata_mos_ru_60562/{filename}")
        list_df.append(df.Cells.apply(pd.Series)[['UNOM', 'ADM_AREA',
↳'DISTRICT', 'SIMPLE_ADDRESS', 'geoData']].assign(BATCH=filename[:-4]).
↳set_index('UNOM'))
    except Exception as exc:
        err_fl.append(int(filename[:6]))
        print('%r generated an exception: %s' % (filename, exc))
        err_fl.append(int(filename[:6]))

with concurrent.futures.ThreadPoolExecutor(max_workers=23) as executor:
    future_to_json = {executor.submit(parse_60562, filename): filename for
↳filename in filelist if filename[:-4] not in set_BATCH}
    for future in tqdm(concurrent.futures.as_completed(future_to_json)):
        pass

time.sleep(.5)
print(f"        {len(list_df)}        ,        : {len(err_fl)}")

if os.path.isfile('../data/UNOM_geoData.pickle'):
    df = pd.read_pickle('../data/UNOM_geoData.pickle')
    df = pd.concat([df]+list_df)
else:
    df = pd.concat(list_df)
df.to_pickle('../data/UNOM_geoData.pickle')

```

```

[ ]: %%time
#                UNOM

import concurrent.futures
import requests
from tqdm import tqdm

api_key = 'd6b9a4bb85fbde55d5cdfa27c94b5566'
url = 'https://apidata.mos.ru/v1/datasets/60562/rows'

```

```
def load_60562(start):
    params = {'api_key': api_key, '$skip': start, '$stop': 500}
    data = ['SIMPLE_ADDRESS', 'UNOM', 'ADM_AREA', 'DISTRICT', 'geoData']
    r = requests.post(url, params=params, json=data)
    with open(f"../data/apidata_mos_ru_60562/{start:06}-{start+499:06}.txt",
        'wb') as f:
        f.write(r.content)
        print(f"../data/apidata_mos_ru_60562/{start:06}-{start+499:06}.txt")

with concurrent.futures.ThreadPoolExecutor(max_workers=5) as executor:
    future_to_file = [executor.submit(load_60562, i) for i in err_fl]
    try:
        for future in tqdm(concurrent.futures.as_completed(future_to_file)):
            pass
    except Exception as e:
        print(e)

# UNOM ( )
```

```
[36]: # , 10
# 2
df_control_10m_pril2 = pd.DataFrame(
    columns = [' ', ' ', ' '],
    data = [
        [' 112', 2303],
        [' ', 2245],
        [' ', ( / )', 1903],
        [' ', 2396],
        [' ( 4 )', 1922],
        [' ( )', 1771],
        [' ', 2096],
        [' ( , , )', 7907],
        [' ', 7906]
    ]
)

df_control_10m_pril2
```

```
[36]: \
0 112
1
2 , ...
3 ,
4 ( ...
5 ( )
6
7 ( , ...
```



8	...
0	2303
1	2245
2	1903
3	2396
4	1922
5	1771
6	2096
7	7907
8	7906

```
[218]: df_pril3 = pd.read_csv('../data/pril3.csv')
df_pril3
```

```
[218]: \
0      39046      2077
1      39045      1853
2      40442      1960
3      39043      2226
4      39042      2258
..      ...      ...
299     38895     38577
300     38708      2150
301     40678      1831
302     39840      2191
303     38710      2175
```

```
\
0      ,
1      ...
2
3      (      ...
4      ,      (      ...
..      ...      ...
299     ,      ,      /      ...
300     ,      ,      ...
301     ,      ,      ...
302
303
```

```
,
0      24
1      72
2      -      -
3      72
```

```

4          12
..      ...
299      72      + .
300      24      + .
301      48
302      -      -
303      72      + .

```

[304 rows x 6 columns]

```

0.1      1. ' ,      . '
      •      « », «      »,      «      ( , ,
      )», «      »      « »
      , « /      »,
      3.

```

[67]: ##

[ ]: #

```

[45]: sum(df[' ' ] == ' ( , , )'), \
      ↪ \
      sum(df[' ' ] == ' '), \
      len(df)

```

[45]: (6743, 21621, 6603578)

```

[46]: sum(df[' ' ] != ' '), sum(df[' ' ] == ' ')

```

[46]: (2931391, 3672187)

```

[54]: df[' ' ].value_counts()

```

```

[54]:      6602735
      829
      12
      2
Name:      , dtype: int64

```

```

[62]: sum(df[' ' ].isin([' ', ' ']))

```

[62]: 6603564

```

[65]: sum(df[' ' ] == ' / ')

```

[65]: 189823

```
[66]: df[''].value_counts()[:10]
```

```
[66]:
768894
667539
474866
368209
255390
192664
189823
180129
176518
147272

/

( )

Name: , dtype: int64
```

```
[81]: # - ' , '
df_bez_vypolnenia = df[
    df[''].isin(['', ''])
    & (~df[''].isin([
        ' ( , , )',
        ' ]))
    & (df[''] != ' ')
    & (df[''] != ' / ')
]
```

```
[82]: # 3
```

```
[ ]: # ,
```

```
[219]: # ,
df_pril3_sroki = df_pril3[['', ' ', ' ', ' ', ' ']].
    ↪ set_index('')
df_pril3_sroki[''].value_counts()
```

```
[219]: - 95
+ . 86
+ 69
48
6
Name: , dtype: int64
```

```
[220]: def_adress_kv = df_pril3_sroki[df_pril3_sroki[''] == ' + .'].index.
    ↪ to_list()
# 10
def_adress_kv[:10]
```

```
[220]: [ ' ( ,
, )',
```

```
[221]: df_defect_kv = df_bez_vypolnenia[df_bez_vypolnenia['  
        ↪isin(def_adress_kv)][  
        ['  
    ] .sort_values(['  
        ↪Timezone'])  
df_defect_kv['  
        ↪Timezone']  
df_defect_kv['  
        (df_defect_kv['  
        &(df_defect_kv['  
        &(df_defect_kv['  
df_defect_kv['  
        ↪Timezone']) // np.timedelta64(1, 'h')  
df_defect_kv
```

12

3838369		,	5	70
4962651		,	55	60
5275267		,	14	104
5570301		,	14	104

	Timezone \
1223439	2021-03-04 10:45:22.742783+03:00
5556596	2021-11-13 13:32:11.641643+03:00
4608695	2021-09-23 15:40:04.137826+03:00
1478730	2021-03-18 20:22:28.938688+03:00
1572687	2021-03-24 18:18:38.735817+03:00
...	...
6469807	2021-12-25 16:51:21.972875+03:00
3838369	2021-08-22 15:24:19.486541+03:00
4962651	2021-10-11 09:39:36.441941+03:00
5275267	2021-10-27 13:27:53.889758+03:00
5570301	2021-11-14 13:32:50.340619+03:00

1223439	NaT	NaN
5556596	NaT	NaN
4608695	NaT	NaN
1478730	2021-03-24 18:18:38.735817+03:00	141.0
1572687	NaT	NaN
...	...	...
6469807	NaT	NaN
3838369	NaT	NaN
4962651	NaT	NaN
5275267	2021-11-14 13:32:50.340619+03:00	432.0
5570301	NaT	NaN

[731893 rows x 6 columns]

```
[222]: df_defect_kv = df_defect_kv.dropna(subset=['', ''])[['', ''],
↳ ' ', ' ', ' ', ' ', ' ']]
df_defect_kv
```

```
[222]:
1478730      ( , )
2786502      ( , )
6467194      ( , )
3978102      ( , )
1947551      ( , )
...
6446440      , , , ( ...
4196341      , , , ( ...
5622471      , , , ( ...
```

```

5658011      ,      ,      ,      ( ...
5275267      ,      ,      ,      ( ...

1478730      1-      ,      7      1
2786502      ,      24,      1      93
6467194      ,      93      183
3978102      ,      6,      2      31
1947551      ,      17      77
...
6446440      ,      105,      1      61
4196341      ,      48      75
5622471      ,      13      145
5658011      ,      13      145
5275267      ,      14      104

```

```

1478730      141.0
2786502      1405.0
6467194      49.0
3978102      4.0
1947551      243.0
...
6446440      72.0
4196341      133.0
5622471      33.0
5658011      5.0
5275267      432.0

```

[119129 rows x 4 columns]

```

[223]: dict_sroki = {x[0]: 0 if x[2]=='-' else int(x[2]) for x in df_pril3_sroki.
        ↪itertuples()}
dict_sroki

```

```

[223]: {'      ,      ': 24,
        ,
        ': 72,
        ,
        ': 0,
        (      )': 72,
        ,      (      )': 12,
        ,      ,      (      )': 6,
        ,      ': 72,
        ,      ,      ': 0,
        ,      (      ,      ,      )': 72,
        ,      ,      (      )': 24,
        ,      (      )': 72,

```

```

'      ,      ' : 72,
'      ,      ,      (      ,
'      )' : 24,
'      /      (      ,      )' : 0,
'      (      )' : 12,
'      ,      ,      (      )' : 24,
'      ,      ,      (      )' : 48,
'      ,      ,      ' : 72,
'      ,      (      ,      ,
'      )' : 12,
'      ,      (      )      ' : 24,
'      ,      ,
'      ' : 48,
'      ,      ' : 24,
'      ' : 24,
'      ,      ,      (      )' : 72,
'      ,      (      )' : 72,
'      ,      ,      (      )' : 72,
'      ,      ,      ,      (
'      )' : 72,
'      ' : 0,
'      (      ,
'      ,      )' : 72,
'      ' : 72,
'      ' : 72,
'      ,      (      ,      )' : 72,
'      ' : 0,
'      (
'      ,      ,      ,      )' : 72,
'      ,      ,      ' : 0,
'      ' : 0,
'      ' : 72,
'      ' : 72,
'      (      ,      )      ' : 72,
'      (      ,      )      ' : 72,
'      (      ,      )      ' : 72,
'      (      )      ' : 0,
'      ,      ,
(      /      )' : 0,
'      ,      ' : 72,
'      ,      ' : 0,
'      ,      ' : 72,
'      ,      ' : 72,
'      ' : 168,
'      ,      ' : 0,
'      ' : 168,
'      (      ,      ,

```

16



```

' : 0,
'
' , , , ' : 0,
' , , , ' : 0,
' ' : 48,
' ( , / ) ' : 24,
' , ( , ) ' : 72,
' ( , ) ' : 72,
' ' : 0,
' ( ) ' : 24,
' ,
( ) ' : 0,
' ' : 72,
' , ( ) ' : 0,
' ' : 72,
' , ( ) ' : 72,
' ( ) ' : 72,
' ' : 72,
' ' : 0,
' , ' : 72,
' , ' : 168,
' , ' : 72,
' , (
) ' : 168,
' , (
) ' : 168,
' , (
) ' : 168,
' , (
) ' : 168,
' ( ) ' : 168,
' , , ,
' : 72,
' ( , ,
) ' : 72,
' , ' : 72,
' ( , ,
) ' : 72,
' , ' : 0,
' ' : 72,
' ( ) ' : 0,
' ( / ,
) ' : 72,
' ' : 0,
' , ' : 72,
' ,
72,
' , ' : 72,

```

```

'      (      ,
)!: 72,
'      /      ,      !: 0,
'      !: 0,
'      (      ,      )!: 72,
'      /      !: 0,
'      (      ,
'      ,      )!: 72,
'      ,      ,      ,      !: 0,
'      ,      ,      !: 0,
'      ,      ,      !: 0,
'      ,      ,      !: 0,
'      ,      ,      !: 48,
'      ,      ,      !: 48,
'      !: 72,
'      /      (      /      )!: 0,
'      !: 0,
'      ,      !: 48,
'      ,      (      )!: 72,
'      ,      (      )!: 72,
'      ,
'      ,      !: 72,
'      ,      ,      (      ,      )!:
72,
'      (      )!: 72,
'      (      )!: 24,
'      (      )!: 72,
'      !: 72,
'      ,      ,      !: 48,
'      ,      !: 48,
'      ,      !: 72,
'      ,      ,      ,      !: 72,
'      ,      ,      ,      !: 0,
'      ,      !: 168,
'      ,      (      )!: 72,
'      ,      !: 72,
'      ,      ,      !: 168,
'      ,      (      )!: 0,
'      ,      ,      !: 72,
'      ,      (      )!: 0,
'      (      )!: 48,
'      !: 72,
'      (      ,      ,      )!: 72,
'      ,      ,      ,      !: 12,
'      /      !: 0,
'      /      ,      !: 0,
'      ,      !: 0,

```

```

'      ,      ,      ,      (      )': 120,
'      ,      ,      ,      ': 72,
'      ,      ,      ,      ': 72,
'      ,      ,      ,      ': 120,
'      ,      ,      ,      ': 72,
'      ,      ,      ,      ,
(      ,      ,      ,      )': 72,
'      ,      ,      ,      (      )': 48,
'      ,      ,      ,      ,      ': 0,
'      ,      ,      ,      ,      ': 0,
'      ,      ,      ,      ,      ': 0,
'      ,      ,      ,      ,      ': 120,
'      ,      ,      ,      ,      ': 0,
'      ,      ,      ,      ,      ': 0,
'      ,      ,      ,      ,      ': 0,
'      ,      ,      ,      ,      ': 0,
'      ,      ,      ,      ,      ': 0,
'      ,      ,      ,      ,      ': 0,
'      ,      ,      ,      ,      ': 0,
'      ,      ,      ,      ,      ': 0,
'      ,      ,      ,      ,      ': 0,
(      ,      )': 120,
'      ,      ,      ,      ,      ': 24,
'      ,      ,      ,      ,      ': 72,
'      ,      ,      ,      ,      (      )': 72,
'      /      ,      ,      ,      (      )': 0,
'      ,      ,      ,      ,      (      )': 48,
'      ,      ,      ,      ,      (      )': 48,
'      ,      ,      ,      ,      (      )': 48,
'      ,      ,      ,      ,      (      )': 24,
'      /      ,      ,      ,      ,      ':
72,
'      /      ,      ,      ,      ,
'      ,      ,      ,      ,      ': 0,
'      ,      ,      ,      ,      (      /      )':
72,
'      ,      ,      ,      ,      -/      (      )': 24,
'      ,      ,      ,      ,      ,      ,
'      ,      ,      ,      ,      )': 120,
'      ,      ,      ,      ,      ,      ': 0,
'      ,      ,      ,      ,      ,      ': 72,
'      ,      ,      ,      ,      ,      ': 0,
'      ,      ,      ,      ,      ,      (      )': 72,
'      ,      ,      ,      ,      ,      ,      ': 24,
'      ,      ,      ,      ,      ,      ,      ': 72,
'      ,      ,      ,      ,      ,      ,      (      )': 24,
'      ,      ,      ,      ,      ,      ,      (      )': 72,
'      ,      ,      ,      ,      ,      ,      (      )': 72,
'      ,      ,      ,      ,      ,      ,      ,      ': 72,

```

```

|
|                                     ,      ,      ' :
120,
|      (      )      ' : 72,
|      (      )' : 72,
|      (      )' : 72,
|      ,      ,      ' : 24,
|      ' : 72,
|      ,      ,      (      )' : 24,
|      ,      (      )' : 72,
|      ' : 72,
|      ,      ,      ' : 120,
|      ' : 72,
|      ' : 24,
|      -      ,      ' : 72,
|      ,      (      ,
|      )' : 72,
|      (      )' : 24,
|      ,      ' : 120,
|      ,      (      )' : 24,
|      (      )' : 120,
|      ,      ,      ' : 48,
|      (      )      ' : 48,
|      ,      (      )      ' : 72,
|      ' : 0,
|      (      ),      ' : 0,
|      ,      ' : 720,
|      ,      ,      ' : 48,
|      ,      ,      (      )' : 0,
|      (      ,      ,      )' : 72,
|      ,      ,      ' : 120,
|      ,      ,
|      (      )' : 120,
|      (      ,      )' : 120,
|      ,      ,      (      )' : 72,
|      ,      ' : 120,
|      ,      ,      ,      ,      (      )' : 168,
|      ,      ,      ' : 48,
|      ' : 0,
|      ,      ' : 72,
|      ,      ' : 0,
|      ,      ,      ' : 6,
|      ,      ,      ,      ' : 48,
|      ,      ,      ,      ' : 48,
|      ,      ' : 48,
|      ' : 48,
|      ' : 48,
|      ' : 48,
|      ,      ' : 48,

```

```

' : 0,
'      (      )' : 24,
'      ' : 0,
'      /      ' : 0,
'      (      )' : 72,
'      ' : 0,
'      ' : 168,
'      ,
( )' : 24,
'      ,      (      )' : 168,
'      ,      ' : 168,
'      ,      (      )' : 168,
'      ,      ,      ( ,      )' : 168,
'      ' : 24,
'      ' : 0,
'      ,      ' : 72,
'      (      )      ,      ' : 72,
'      (      ,
'      ,      ,      )' : 72,
'      ,      ,      ' : 72,
'      ,      ,      (      )' : 72,
'      ,      ,      ' : 48,
'      ,      ,      (      )' : 72,
'      ,      ,      (      ,      ,      )' : 48,
'      ,      ,      (      ,      )' : 48,
'      ,      (      )' : 120,
'      ,      (      )' : 24,
'      ,      ,      (      )' : 0,
'      ,      (      )' : 24,
'      ,      (      )' :
24,
'      ,      ' : 48,
'      ,      ,      (      ,      ,
'      )' : 48,
'      ,      (      ,      )' : 48,
'      ,      ,      ' : 9,
'      ' : 24,
'      ,      ' : 6,
'      ,      ' : 48,
'      ,
'      ' : 0,
'      ,      ,      ' : 72,
'      ,      ,      ' : 0,
'      ,      ,      ' : 72,
'      ,      ,      ' : 72,
'      ,      ,      /      ' : 72,
'      ,      ,      ' : 24,

```

```

',
',
': 0,
',
': 72}
': 48,

```

```
[226]: df_defect_kv[''].apply(lambda x: 0 if x not in dict_sroki else dict_sroki[x])
```

```
[226]: 1478730      72
      2786502      72
      6467194      72
      3978102      72
      1947551      72
      ...
      6446440     168
      4196341     168
      5622471     168
      5658011     168
      5275267     168
      Name: , Length: 119129, dtype: int64
```

```
[227]: df_defect_kv[''] = df_defect_kv[''].apply(lambda x: 0 if x not in dict_sroki else dict_sroki[x])
df_defect_kv
```

```
[227]:
1478730      ( , ) \
2786502      ( , )
6467194      ( , )
3978102      ( , )
1947551      ( , )
...
6446440      , , , ( ...
4196341      , , , ( ...
5622471      , , , ( ...
5658011      , , , ( ...
5275267      , , , ( ...

1478730      1- , 7 1 \
2786502      , 24, 1 93
6467194      , 93 183
3978102      , 6, 2 31
1947551      , 17 77
...
6446440      , 105, 1 61
4196341      , 48 75
5622471      , 13 145
```

5658011	,	13	145
5275267	,	14	104

1478730	141.0	72
2786502	1405.0	72
6467194	49.0	72
3978102	4.0	72
1947551	243.0	72
...	...	...
6446440	72.0	168
4196341	133.0	168
5622471	33.0	168
5658011	5.0	168
5275267	432.0	168

[119129 rows x 5 columns]

```
[245]: df_defect_v1_kv = df_defect_kv[df_defect_kv['
↳      ']<df_defect_kv['      ']]
df_defect_v1_kv
```

```
[245]:
6467194      ( , )
3978102      ( , )
6310389      ( , )
2195731      ( , )
3727304      ( , )
...
4822743      , , , ( ...
6446440      , , , ( ...
4196341      , , , ( ...
5622471      , , , ( ...
5658011      , , , ( ...
```

6467194			,	93	183
3978102			, 6,	2	31
6310389			, 10,	6	105
2195731				, 86	122
3727304			, 5,	4	21
...				...	...
4822743				, 29	11
6446440			, 105,	1	61
4196341				, 48	75
5622471				, 13	145
5658011				, 13	145

6467194	49.0	72
3978102	4.0	72
6310389	19.0	72
2195731	1.0	72
3727304	42.0	72
...	...	...
4822743	47.0	168
6446440	72.0	168
4196341	133.0	168
5622471	33.0	168
5658011	5.0	168

[70474 rows x 5 columns]

```
[249]: # Top20
#
df_defect_v1_kv.value_counts('1', '2', '3', '4', '5')[:20]
```

[249]:

11851	,
4302	,
4008	-
3962	,
3470	( )
3079	( )
2965	( )
2564	( )
2491	( )
1821	
1701	
1646	( )
1378	



```

( / )
1325
1285
1276
, , , , ( ,
) 1156
, ( , )
1067
, , ( 909
, ,
863
dtype: int64

```

```
[237]: df.loc[df_defect_v1.index]
```

```

[237]:
\
6467194      106193182      106387164  06470407/21
3978102      93839045      93839641  03981226/21
6310389      105419968      105421743  06313572/21
2195731      84201625      84207220  02197540/21
3727304      92519600      92520371  03730410/21
...
4822743      97936434      97937042  04825885/21
6446440      106092363      106123047  06449630/21
4196341      94973757      95035865  04199466/21
5622471      101965021      101966636  05625652/21
5658011      102147506      102166671  05661192/21

```

```

(mos.ru) \
6467194      NaN
3978102      NaN
6310389      0002-9000005-207306-0911653/21
2195731      0002-9000005-207306-0270277/21
3727304      NaN
...
4822743      NaN
6446440      NaN
4196341      NaN
5622471      NaN
5658011      NaN

```

```

Timezone \
6467194      2021-12-25 15:10:54.173380+03:00
3978102      2021-08-31 08:59:17.928396+03:00

```

6310389	2021-12-19 14:04:05.599346+03:00
2195731	2021-05-03 14:47:56.594351+03:00
3727304	2021-08-15 14:15:39.356242+03:00
...	...
4822743	2021-10-03 14:16:19.029216+03:00
6446440	2021-12-24 16:15:17.441769+03:00
4196341	2021-09-11 22:57:52.742430+03:00
5622471	2021-11-17 08:21:55.548379+03:00
5658011	2021-11-18 17:36:25.987858+03:00

	Timezone \
6467194	2021-12-27 13:54:02.142282+03:00
3978102	2021-08-31 09:05:22.758503+03:00
6310389	2021-12-19 14:18:25.864326+03:00
2195731	2021-05-03 16:17:54.718514+03:00
3727304	2021-08-15 14:28:45.538033+03:00
...	...
4822743	2021-10-03 14:24:37.206186+03:00
6446440	2021-12-24 20:21:12.856319+03:00
4196341	2021-09-13 07:59:11.941390+03:00
5622471	2021-11-17 08:42:41.004113+03:00
5658011	2021-11-18 21:09:42.436697+03:00

\
   
 6467194
   
 3978102
   
 6310389
   
 2195731
   
 3727304
   
 ...
   
 4822743
   
 6446440
   
 4196341
   
 5622471
   
 5658011

	\	
6467194	phone	frontline
3978102	phone	frontline
6310389	mobile	emp2
2195731	mobile	emp2
3727304	phone	frontline
...	...	...
4822743	phone	frontline
6446440	phone	frontline
4196341	elec_house	ElecH
5622471	elec_house	ElecH

5658011	elec_house	ElecH
---------	------------	-------

	...		\
6467194	...	2021-12-27 13:54:02.142282+03:00	
3978102	...	2021-08-31 09:05:22.758503+03:00	
6310389	...	2021-12-19 14:18:25.864326+03:00	
2195731	...	2021-05-03 16:17:54.718514+03:00	
3727304	...	2021-08-15 14:28:45.538033+03:00	
...	...	...	...
4822743	...	2021-10-03 14:24:37.206186+03:00	
6446440	...	2021-12-24 20:21:12.856319+03:00	
4196341	...	2021-09-13 07:59:11.941390+03:00	
5622471	...	2021-11-17 08:42:41.004113+03:00	
5658011	...	2021-11-18 21:09:42.436697+03:00	

			\
6467194	2021-12-25 16:00:00+03:00	2021-12-25 20:00:00+03:00	
3978102	2021-08-31 09:00:00+03:00	2021-08-31 14:00:00+03:00	
6310389	2021-12-20 06:00:00+03:00	2021-12-20 12:00:00+03:00	
2195731	2021-05-03 14:00:00+03:00	2021-05-03 20:00:00+03:00	
3727304	2021-08-15 15:00:00+03:00	2021-08-15 20:00:00+03:00	
...	...	...	...
4822743	2021-10-03 15:00:00+03:00	2021-10-03 20:00:00+03:00	
6446440	2021-12-24 17:00:00+03:00	2021-12-24 20:00:00+03:00	
4196341	2021-09-12 11:00:00+03:00	2021-09-12 20:00:00+03:00	
5622471	2021-11-17 09:00:00+03:00	2021-11-17 18:00:00+03:00	
5658011	2021-11-19 13:00:00+03:00	2021-11-19 17:00:00+03:00	

	/	\
6467194	NaT	NaN
3978102	NaT	NaN
6310389	NaT	NaN
2195731	NaT	NaN
3727304	NaT	NaN
...	...	...
4822743	NaT	NaN
6446440	NaT	NaN
4196341	2021-09-14 18:26:44.350336+03:00	NaN
5622471	2021-11-17 17:30:53.667474+03:00	NaN
5658011	NaT	NaN

		\	
6467194	NaN		NaN
3978102	NaN		NaN
6310389	NaN		NaN
2195731	NaN		NaN
3727304	NaN		NaN

6467194	
3978102	
6310389	
2195731	
3727304	
...	...
4822743	
6446440	
4196341	
5622471	
5658011	

```
[239]: #
def_adress_pod = df_pril3_sroki[df_pril3_sroki[''] == ' ' + ''].index.
↳to_list()
# 10
def_adress_pod[:10]
```

28

$$\begin{aligned} & \quad (\quad , \quad / \quad )' , \\ & \quad , \quad , \\ & \quad , \\ & \quad )' , \\ & \quad , \\ & \quad )' , \\ & \quad ( \quad )'] \end{aligned}$$
[illegible]

```
[242]:
```

1558065	,	1-	,	4
1649215	,	1-	,	4
6383442	,	1-	, 7,	1
6383526	,	1-	, 7,	1
5931030	,		1-	, 7
...		...		...
6142346	,		, 6,	1
6260875	,		, 19,	1
4240041	,			31
4245294	,			31
4949659	,		, 7,	1
Timezone \				
1558065	12	2021-03-23	21:00:41.984148+03:00	
1649215	12	2021-03-29	16:07:33.808503+03:00	
6383442	1	2021-12-22	12:33:11.397789+03:00	
6383526	1	2021-12-22	12:35:12.487604+03:00	
5931030	3	2021-12-02	17:28:35.174187+03:00	
...	...		...	
6142346	1	2021-12-12	15:15:44.212224+03:00	
6260875	1	2021-12-17	08:05:11.512369+03:00	
4240041	1	2021-09-14	13:07:55.303234+03:00	
4245294	1	2021-09-14	16:17:58.100563+03:00	

4949659            5            2021-10-10 11:06:30.017322+03:00

1558065	2021-03-29 16:07:33.808503+03:00	139.0
1649215	NaT	NaN
6383442	2021-12-22 12:35:12.487604+03:00	0.0
6383526	NaT	NaN
5931030	NaT	NaN
...	...	...
6142346	NaT	NaN
6260875	NaT	NaN
4240041	2021-09-14 16:17:58.100563+03:00	3.0
4245294	NaT	NaN
4949659	NaT	NaN

[336490 rows x 6 columns]

```
[243]: df_defect_pod = df_defect_pod.dropna(subset=['', ''])[['', ''],
df_defect_pod
```

```
[243]: \
1558065      ,      1-      ,      4
6383442      ,      1-      ,      7,      1
3253988      ,      15-      ,      45
3655982      ,      15-      ,      45
3678192      ,      15-      ,      45
...
4308859      ,      ,      8
4321713      ,      ,      8
4330529      ,      ,      8
4330950      ,      ,      8
4240041      ,      ,      31
```

1558065	12	139.0
6383442	1	0.0
3253988	2	659.0
3655982	2	38.0
3678192	2	315.0
...	...	...
4308859	1	15.0
4321713	1	1.0
4330529	1	0.0
4330950	1	3.0
4240041	1	3.0

[132823 rows x 4 columns]

```
[244]: df_defect_pod['      '] = df_defect_pod['      '].apply(lambda x: 0 if x_
↳ not in dict_sroki else dict_sroki[x])
df_defect_pod
```

[244]: \

1558065	,	1-	,	4
6383442	,	1-	, 7,	1
3253988	,	15-	,	45
3655982	,	15-	,	45
3678192	,	15-	,	45
...		...		...
4308859	,		,	8
4321713	,		,	8
4330529	,		,	8
4330950	,		,	8
4240041	,		,	31

1558065	12	139.0	720
6383442	1	0.0	720
3253988	2	659.0	720
3655982	2	38.0	720
3678192	2	315.0	720
...	...	...	...
4308859	1	15.0	24
4321713	1	1.0	24
4330529	1	0.0	24
4330950	1	3.0	24
4240041	1	3.0	24

[132823 rows x 5 columns]

```
[247]: df_defect_v1_pod = df_defect_pod[df_defect_pod['      '
↳      ']<df_defect_pod['      ']]
df_defect_v1_pod
```

[247]: \

1558065	,	1-	,	4
6383442	,	1-	, 7,	1
3253988	,	15-	,	45
3655982	,	15-	,	45
3678192	,	15-	,	45
...		...		...
4308859	,		,	8
4321713	,		,	8

4330529				8
4330950				8
4240041				31

1558065	12		139.0	720
6383442	1		0.0	720
3253988	2		659.0	720
3655982	2		38.0	720
3678192	2		315.0	720
...	...		...	...
4308859	1		15.0	24
4321713	1		1.0	24
4330529	1		0.0	24
4330950	1		3.0	24
4240041	1		3.0	24

[45181 rows x 5 columns]

```
[248]: # Top20
#
df_defect_v1_pod.value_counts('1', 1, 1)[:20]
```

```
[248]:
6816
4466
4462
4096
1869
1685
1666
1197
1130
1095
```



```

1029                                     (      ,      )
893
885
661      ,      ,      ,      ,
576                                     (      ,      ,      )
502      ,      ,
      (      )      502
468      ,
465      ,      ,      (      )
452      (      ,      ,      )
dtype: int64

```

```
[252]: #
def_adress = df_pril3_sroki[df_pril3_sroki['
    ↪index.to_list()
# 10
def_adress[:10]
```

```
[252]: [ '          ,      ',  
        '              ,  
            ',  
            '          (         )',  
            '       (           )',  
            '    ,     ,      (         )',  
            '   ,             ',  
            '    ,          (           , , )',  
            '                (           )',  
            '          ,      (         )',  
            '               ,      ']
```

```
[257]: df_defect_addr = df_bez_vypolnenia[df_bez_vypolnenia['  
        ↪isin(def_address)][  
            ['  
                ], '  
                ], '  
                Timezone']  
].sort_values(['  
                ', '  
                ', '  
                Timezone'])  
df_defect_addr['  
                '] = df_defect_addr.shift(-1)['  
                ]  
        ↪Timezone']  
df_defect_addr['  
                '] = df_defect_addr['  
                '].where(  
    (df_defect_addr['  
                ']==df_defect_addr['  
                '].shift(-1))  
    &(df_defect_addr['  
                ']==df_defect_addr['  
                '].shift(-1)), np.nan)
```

```
df_defect_addr[''] = (df_defect_addr[''] -
    df_defect_addr['Timezone']) // np.timedelta64(1, 'h')
df_defect_addr
```

[257]:

```
2259666      ( , )
1684367      ( , )
1736458      ( , )
3188206      ( , )
3572524      ( , )
...
4778809      ,
5709243      ,
4562283      ,
6224168      ,
6599361      ,
```

```
2259666      1-      ,      6
1684367  1-      ,      21,      1
1736458  1-      ,      21,      1
3188206  1-      ,      21,      1
3572524  1-      ,      22,      1
...
4778809      ,      8,      2
5709243      ,      5,      1
4562283      ,      43,      2
6224168      ,      3,      3
6599361      ,      4,      24
```

```
Timezone \
2259666      2021-05-09 10:45:37.824920+03:00
1684367      2021-03-31 14:48:12.215993+03:00
1736458      2021-04-03 16:54:41.231484+03:00
3188206      2021-07-09 10:06:43.530347+03:00
3572524      2021-08-05 08:31:29.044229+03:00
...
4778809      2021-09-30 20:59:29.604804+03:00
5709243      2021-11-21 17:28:28.874107+03:00
4562283      2021-09-22 12:40:16.436932+03:00
6224168      2021-12-15 16:41:18.712985+03:00
6599361      2021-12-31 15:02:41.639664+03:00
```

```
2259666      NaT      NaN
1684367  2021-04-03 16:54:41.231484+03:00      74.0
1736458  2021-07-09 10:06:43.530347+03:00      2321.0
```

3188206	NaT	NaN
3572524 2021-08-16 09:23:24.004233+03:00		264.0
...	...	...
4778809	NaT	NaN
5709243	NaT	NaN
4562283	NaT	NaN
6224168	NaT	NaN
6599361	NaT	NaN

[111058 rows x 5 columns]

```
[258]: df_defect_addr = df_defect_addr.dropna(subset=['', ''])[['', '']]
df_defect_addr
```

```
[258]: \
1684367      ( , )
1736458      ( , )
3572524      ( , )
37063        ( , )
3256722      ( , )
...
4071046      ,
4072245      ,
4074991      ,
4113028      ,
4113159      ,
...
1684367  1-      , 21, 1
1736458  1-      , 21, 1
3572524  1-      , 22, 1
37063      10-      , 20
3256722      16-      , 18
...
4071046      , 9, 1
4072245      , 9, 1
4074991      , 9, 1
4113028      , 9, 1
4113159      , 9, 1
```

1684367	74.0
1736458	2321.0
3572524	264.0
37063	3289.0
3256722	0.0

```
...
4071046      1.0
4072245      2.0
4074991     43.0
4113028      0.0
4113159     25.0
```

[53092 rows x 3 columns]

```
[262]: df_defect_addr['      '] = df_defect_addr['      '].apply(lambda x: 0 if x_
↳not in dict_sroki else dict_sroki[x])
df_defect_addr
```

[262]:

```
1684367      ( , )
1736458      ( , )
3572524      ( , )
37063        ( , )
3256722      ( , )
```

```
...
4071046      ,
4072245      ,
4074991      ,
4113028      ,
4113159      ,
```

```
1684367  1-      ,  21,  1
1736458  1-      ,  21,  1
3572524  1-      ,  22,  1
37063      10-    ,  20
3256722      16-    ,  18
```

```
...
4071046      ,  9,  1
4072245      ,  9,  1
4074991      ,  9,  1
4113028      ,  9,  1
4113159      ,  9,  1
```

```
1684367      74.0      72
1736458     2321.0     72
3572524     264.0     72
37063      3289.0     72
3256722       0.0     72
```

```
...
4071046      1.0      48
```

4072245	2.0	48
4074991	43.0	48
4113028	0.0	48
4113159	25.0	48

[53092 rows x 4 columns]

```
[263]: df_defect_v1_addr = df_defect_addr[df_defect_addr['
↪      ']<df_defect_addr['      ']]
df_defect_v1_addr
```

```
[263]: \
3256722      ( , )
4253186      ( , )
456784      ( , )
497558      ( , )
500662      ( , )
...
4071046      ,
4072245      ,
4074991      ,
4113028      ,
4113159      ,

\
3256722      16-      ,      18      0.0
4253186      2-      ,      4      0.0
456784      9-      ,      66,      2      49.0
497558      9-      ,      66,      2      3.0
500662      9-      ,      66,      2      14.0
...
4071046      ,      9,      1      1.0
4072245      ,      9,      1      2.0
4074991      ,      9,      1      43.0
4113028      ,      9,      1      0.0
4113159      ,      9,      1      25.0

3256722      72
4253186      72
456784      72
497558      72
500662      72
...
4071046      48
4072245      48
4074991      48
```

```
4113028      48
4113159      48
```

```
[17924 rows x 4 columns]
```

```
[264]: # Top20
#
df_defect_v1_addr.value_counts('1 ' , . '')[20]
```

```
[264]:
```

```
6745
2664
1842
1678
911
492
343
302
262
216
210
208
206
179
166
158
124
109
```

```

99
, , ( )
94
dtype: int64

```

```
[266]: df_defect_v1_kv.index
```

```
[266]: Int64Index([6467194, 3978102, 6310389, 2195731, 3727304, 2268729, 2296875,
3569825, 1320696, 1781511,
...
4991760, 5001573, 6334893, 6406808, 6447511, 4822743, 6446440,
4196341, 5622471, 5658011],
dtype='int64', length=70474)
```

```
[267]: # 1
index_template1 = list(df_defect_v1_kv.index) + list(df_defect_v1_pod.index) +
↳ list(df_defect_v1_addr.index)
len(index_template1)
```

```
[267]: 133579
```

```
[268]: # 1 ' , . ' ( )
df.loc[index_template1]
```

```
[268]:
6467194      106193182      106387164  06470407/21
3978102      93839045      93839641  03981226/21
6310389      105419968      105421743  06313572/21
2195731      84201625      84207220  02197540/21
3727304      92519600      92520371  03730410/21
...
4071046      94321521      94326726  04074170/21
4072245      94327196      94340052  04075369/21
4074991      94340541      94346874  04078115/21
4113028      94530559      94584031  04116153/21
4113159      94531234      94536269  04116284/21
```

```

(mos.ru) \
6467194      NaN
3978102      NaN
6310389      0002-90000005-207306-0911653/21
2195731      0002-90000005-207306-0270277/21
3727304      NaN
...
4071046      NaN
4072245      NaN
4074991      NaN

```

4113028	NaN
4113159	NaN

	Timezone \
6467194	2021-12-25 15:10:54.173380+03:00
3978102	2021-08-31 08:59:17.928396+03:00
6310389	2021-12-19 14:04:05.599346+03:00
2195731	2021-05-03 14:47:56.594351+03:00
3727304	2021-08-15 14:15:39.356242+03:00
...	...
4071046	2021-09-05 14:17:12.863957+03:00
4072245	2021-09-05 15:29:59.513757+03:00
4074991	2021-09-05 18:25:44.990224+03:00
4113028	2021-09-07 13:42:01.452953+03:00
4113159	2021-09-07 13:46:25.101260+03:00

	Timezone \
6467194	2021-12-27 13:54:02.142282+03:00
3978102	2021-08-31 09:05:22.758503+03:00
6310389	2021-12-19 14:18:25.864326+03:00
2195731	2021-05-03 16:17:54.718514+03:00
3727304	2021-08-15 14:28:45.538033+03:00
...	...
4071046	2021-09-05 15:23:47.964751+03:00
4072245	2021-09-05 18:19:06.969856+03:00
4074991	2021-09-05 20:07:51.752166+03:00
4113028	2021-09-07 20:56:11.676850+03:00
4113159	2021-09-07 14:21:28.585247+03:00

	\
6467194	
3978102	
6310389	
2195731	
3727304	
...	...
4071046	
4072245	
4074991	
4113028	
4113159	

	\	
6467194	phone	frontline
3978102	phone	frontline
6310389	mobile	emp2
2195731	mobile	emp2



3727304	phone	frontline
...	...	...
4071046	phone	voxcom
4072245	phone	frontline
4074991	phone	frontline
4113028	phone	frontline
4113159	phone	frontline

	...		\
6467194	...	2021-12-27	13:54:02.142282+03:00
3978102	...	2021-08-31	09:05:22.758503+03:00
6310389	...	2021-12-19	14:18:25.864326+03:00
2195731	...	2021-05-03	16:17:54.718514+03:00
3727304	...	2021-08-15	14:28:45.538033+03:00
...	...	...	...
4071046	...	2021-09-05	15:23:47.964751+03:00
4072245	...	2021-09-05	18:19:06.969856+03:00
4074991	...	2021-09-05	20:07:51.752166+03:00
4113028	...	2021-09-07	20:56:11.676850+03:00
4113159	...	2021-09-07	14:21:28.585247+03:00

			\
6467194	2021-12-25	16:00:00+03:00	2021-12-25 20:00:00+03:00
3978102	2021-08-31	09:00:00+03:00	2021-08-31 14:00:00+03:00
6310389	2021-12-20	06:00:00+03:00	2021-12-20 12:00:00+03:00
2195731	2021-05-03	14:00:00+03:00	2021-05-03 20:00:00+03:00
3727304	2021-08-15	15:00:00+03:00	2021-08-15 20:00:00+03:00
...	...	...	...
4071046		NaT	NaT
4072245		NaT	NaT
4074991		NaT	NaT
4113028		NaT	NaT
4113159		NaT	NaT

	/		\
6467194	NaT	NaN	NaN
3978102	NaT	NaN	NaN
6310389	NaT	NaN	NaN
2195731	NaT	NaN	NaN
3727304	NaT	NaN	NaN
...	...	...	...
4071046	NaT	NaN	NaN
4072245	NaT	NaN	NaN
4074991	NaT	NaN	NaN
4113028	NaT	NaN	NaN
4113159	NaT	NaN	NaN

6467194	NaN
3978102	NaN
6310389	NaN
2195731	NaN
3727304	NaN
...	...
4071046	NaN
4072245	NaN
4074991	NaN
4113028	NaN
4113159	NaN

6467194	free
3978102	free
6310389	free
2195731	free
3727304	free
...	...
4071046	free
4072245	free
4074991	free
4113028	free
4113159	free

[133579 rows x 68 columns]

```
[270]: df.loc[index_template1].to_pickle('../data/anomalii_1.pickle')
```

```
[ ]:
```