Чепоков Елизар ПИ-18-2

1. **Краткая информация о проекте**. Технологический процесс добычи нефти. В работе находятся несколько скважин, каждая из которых оснащена насосным оборудованием. Оборудование может работать в различных режимах. Основные технологические параметры о работе оборудования измеряются с дискретностью 1 сутки (раз в 24 часа). В связи с нестабильной связью, особенностями сбора данных (ручной ввод) имеются значительные пропуски в данных
2. **Исходные данные** - файл Excel с параметрами скважин, результатами измерений, режимами работы оборудования. Каждая вкладка файла содержит данные по 1 месяцу работы всего массива скважин.
3. "**Шпаргалка**" для анализа данных
    - Обработка данных для исследований.py - реализует парсинг данных из исходного файла и формирование датафрейма. Формируется новый датафрейм с данными только по одной выбранной скважине. Также в файле реализуется начальная очистка данных (удаление столбцов с пустыми значениями во всех строках) для выбранной скважины
    - Сохранение_данных_в_CSV_для_Influx.py - реализует парсинг данных из исходного файла, подготовку датафрейма для экспорта и сам экспорт в csv файл. Далее можно использоваться скрипт CSV2Influx.py для экспорта в БД InfluxDB

НИС №7 "Работа по проекту анализа данных IoT в группах"

```
# -*- coding: utf-8 -*-
"""
Created on Tue May  7 22:29:48 2019

@author: пк
"""

# в консоли IPython      ->      !pip install influxdb
from influxdb import InfluxDBClient
import pandas as pd
import numpy as np

data = 'Дата замера'
hole = 'Скважина'
x1 = 'Способ эксплуатации'
x2 = 'Режим'
y1 = 'Рпр(ТМ)'
y2 = 'Рзаб(Рпр)'
y3 = 'Рзаб(Нд)'
y4 = 'Рзаб(иссл)'

file_name = 'Данные для исследований.xlsx'

def read_all_sheets(file_name_excel):
    df = pd.DataFrame()
    xls = pd.ExcelFile(file_name_excel)
    for list_excel in xls.sheet_names:
        df = df.append(pd.read_excel(xls, list_excel, parse_dates=[data], index_col=data))
    return df

df = read_all_sheets(file_name)
df.sort_index(inplace=True)

def replace_text_values_in_x(df, nameX):
```

```
    dict_changes = {}
    _list = pd.unique(df[nameX]).tolist()
    i = 1
    for value in _list:
        if (str(value) != str(np.NaN)):
            df.loc[df[nameX] == value, nameX] = i
            dict_changes[i] = value
            i += 1
    df[nameX] = df[nameX].fillna(len(_list))
    dict_changes[len(_list)] = np.NaN
    return dict_changes

what_replaced_x1 = replace_text_values_in_x(df, x1)
what_replaced_x2 = replace_text_values_in_x(df, x2)

all_data = df.copy()

cleaning_map = lambda x: str(x).strip()
all_data[hole] = all_data[hole].map(cleaning_map)

all_df_to_influx = all_data.copy()[[hole, x1, x2, y1, y2, y3, y4]]

influx_file_name = '_data_2018_01-07'
list_of_holes = pd.unique(all_data[hole]).tolist()
count_empty_data = 0
list_empty_data = []
cleaned_data = pd.DataFrame()
for _hole in list_of_holes:
    df_to_influx = all_df_to_influx[all_df_to_influx[hole] == _hole][[x1, x2, y1, y2, y3, y4]]
    df_to_influx.insert(loc=0, column='Время', value=df_to_influx.index.time[0])
    temp_df = df_to_influx[[y1, y2, y3, y4]].dropna(axis=0, how='all')
    if (not temp_df.empty):
        cleaned_data = cleaned_data.append(df_to_influx)
        df_to_influx.to_csv('output\\' + str(_hole) + influx_file_name+'.csv', encoding='cp1251',
sep=';')
    else:
        #print('Для скважины '+ _hole +' нет данных по давлению!')
        list_empty_data.append(_hole)
        count_empty_data += 1
print('Количество скважен без данных по давлению: '+ str(count_empty_data))
print('Список скважен без данных по давлению:')
i=1
for _hole in list_empty_data:
    print(str(i)+'. '+_hole)
    i+=1
print('Сохранение в отдельные файлы ".csv" выполнено!')

"""
writer = pd.ExcelWriter(influx_file_name+'.xlsx', engine='xlsxwriter',
datetime_format='dd.mm.yyyy', date_format='dd.mm.yyyy')
df_to_influx.to_excel(writer)
writer.save()

ex = df_to_influx.dropna()
writer = pd.ExcelWriter(influx_file_name+'1.xlsx', engine='xlsxwriter',
datetime_format='dd.mm.yyyy', date_format='dd.mm.yyyy')
ex.to_excel(writer)
```

```
writer.save()
"""
#ex1 = all_data.copy()[['Скважина', 'Режим', 'Способ эксплуатации', 'Рзаб(иссл)']].dropna()

# making connection to influx
#client = InfluxDBClient(host='localhost', port=8086, username='myuser', password='mypass'
ssl=True, verify_ssl=True)


# -*- coding: utf-8 cp1251 -*-
"""
Created on 0 0:00:00 0000

@author: Bloodies
"""

# !pip install influxdb
from influxdb import InfluxDBClient
import pandas as pd
import numpy as np

data = 'Дата замера'
hole = 'Скважина'
x1 = 'Способ эксплуатации'
x2 = 'Режим'
y1 = 'Рпр(ТМ)'
y2 = 'Рзаб(Рпр)'
y3 = 'Рзаб(Нд)'
y4 = 'Рзаб(иссл)'

file_name = 'data.xlsx'

def read_all_sheets(file_name_excel):
    df = pd.DataFrame()
    xls = pd.ExcelFile(file_name_excel)
    for list_excel in xls.sheet_names:
        df = df.append(pd.read_excel(xls, list_excel, parse_dates=[data], index_col=data))
    return df

df = read_all_sheets(file_name)
df.sort_index(inplace=True)

def replace_text_values_in_x(df, nameX):
    dict_changes = {}
    _list = pd.unique(df[nameX]).tolist()
    i = 1
    for value in _list:
        if (str(value) != str(np.NaN)):
            df.loc[df[nameX] == value, nameX] = i
            dict_changes[i] = value
            i += 1
    df[nameX] = df[nameX].fillna(len(_list))
    dict_changes[len(_list)] = np.NaN
    return dict_changes

what_replaced_x1 = replace_text_values_in_x(df, x1)
what_replaced_x2 = replace_text_values_in_x(df, x2)
```

```
all_data = df.copy()

cleaning_map = lambda x: str(x).strip()
all_data[hole] = all_data[hole].map(cleaning_map)
all_df_to_influx = all_data.copy()[[hole, x1, x2, y1, y2, y3, y4]]

influx_file_name = '_data_2018_01-07'
list_of_holes = pd.unique(all_data[hole]).tolist()
count_empty_data = 0
fill_method = 'bfill'
list_empty_data = []
cleaned_data = pd.DataFrame()
for _hole in list_of_holes[1:100]:
    if count_empty_data > 0:
        fill_method = 'ffill'
    df_to_influx = all_df_to_influx[all_df_to_influx[hole] == _hole][[x1, x2, y1, y2, y3, y4]]
    df_to_influx.insert(loc=0, column='Время', value=df_to_influx.index.time[0])
    df_to_influx.insert(loc=0, column='Скважина', value=_hole)
    temp_df = df_to_influx[[y1, y2, y3, y4]].dropna(axis=1, how='all')
    if (not temp_df.empty):
        cleaned_data = cleaned_data.append(df_to_influx)
        #cleaned_data[y1].fillna(method=fill_method, inplace=True)
        #cleaned_data[y2].fillna(method=fill_method, inplace=True)
        #cleaned_data[y2].fillna(method=fill_method, inplace=True)
        #cleaned_data[y3].fillna(method=fill_method, inplace=True)
        #cleaned_data[y4].fillna(method=fill_method, inplace=True)

        cleaned_data.to_csv('output\\' + 'holes'+'.csv', encoding='cp1251', sep=';')
    else:
        list_empty_data.append(_hole)
        count_empty_data += 1

print('Сохранение в файл ".csv" выполнено!')
```