LUMEN®

Parsing “orderData” to JSON object

Abstract

This Python script processes data from an Excel file's "Baseline_Request" column containing a key named 'orderData' representing order information. Each orderData key's value is parsed to extract specific fields and organize them into a structured JSON format.

Prepared By

Vaishnavi Kulkarni, under the guidance of Mohit Vaishnav.

Revision History

Date	Revision	Author	Description
03/20/2024	0.1	Vaishnavi	Parse orderData

Contents

Problem Statement

Given a sample excel file that contains a column named ‘Baseline_Request’ and each row of this column contains a key ‘orderData’ . The requirement was to write a code which takes orderData as input and parses it into a JSON object for different sections based on certain specifications.

Data Needed

1. An excel file containing ‘Baseline_Request’ column.

2. The ‘orderData’ key containing value which needs to be parsed into JSON format.

Solution

The Pandas python library is used to work with the datasets as this library has functions for analyzing, cleaning and manipulating data and the DataFrame object of Pandas library is used in the code to access and work with the excel data.

In the code, a function "parse_text" is created to process the text input line by line, splitting it into sections and variables. It organizes this data into a structured format where each section contains variables with associated details like sequence number, parent, varType, action, varName and varValue and lines with variable type ' ' or 'N' append their values to the previous line's value.

The logic used to segregate the orderData text into different section is given below -

```
def parse_text(text):
    lines = text.strip().split('\n')
    parsed_data = {}
    current_section = None
    current_var_type = None
    for line in lines:
        if len(line) < 21:
            continue
        sequenceNumber = line[0:6]
        parent = line[6:12]
        sectionNumber = line[12:14]
        varType = line[14]
        action = line[15]
        varName = line[16:21]
        varValue = line[21:].strip()

        if varType == 'S':
            current_section = varName
            current_var_type = None
            parsed_data[current_section] = []

        if current_section is not None and current_var_type not in {' ', 'N'}:
            if varType in {' ', 'N'}:
                if parsed_data[current_section]:
                    parsed_data[current_section][-1]['varValue'] += ' ' + varValue
            else:
                parsed_data[current_section].append({
                    'sqNo': sequenceNumber.strip(),
                    'parent': parent.strip(),
                    'sectionNo': sectionNumber.strip(),
                    'varType': varType.strip(),
                    'action': action.strip(),
                    'varName': varName.strip(),
                    'varValue': varValue.strip()
                })

        if varType not in {' ', 'N'}:
            current_var_type = varType

    return parsed_data
```

Team/Members Involved

K V Gopalakrishna

Deepak K Dubey

Mohit Vaishnav

Vaishnavi Kulkarni

Data Collection

A sample excel sheet that had to be taken as input data was provided.

orderdata

The Baseline_Request column in this file had around 1746 rows , each containing orderData that had to be parsed.

Sample input

{"orderData": "00000100000000S STARTD33195471\n00000200000000X ORDSTAO\n00000300000000X
FUNC PRE\n00000400000000X STCD AZ\n00000500000000X ENTOPTOMDMN\n00000600000000X
ENTDT2023-08-06\n00000700000000X ENTTM03.36.03\n00000800000000X
MODOPTOMDMN\n00000900000000X MODDT2023-08-06\n00001000000000X
MODTM03.36.03\n00001100000000X DISDT23218\n00001200000000X
DISTM0336\n00001300000000X SECNT0047\n00001400000000X WRCEN928683\n00001500000000X
ENTCDF\n00001600000000X TYPINR\n00001700000000X TYPOTR\n00001800000000X DDCD
X\n00001900000000X OWNRS6\n00002000000000X MODAP6\n00002200000002S HDR
\n00002300000002L SRC AO\n00002400000002L RELDT2023-08-06\n00002500000003S ID
\n00002600000003L TN   928 683-2049\n00002700000003F CUS 278\n00002800000003F EX
GBD\n00002900000003L CS   GFR\n00003000000003L ORDT D\n00003100000003L ORDN
33195471\n00003200000003F APPT X\n00003300000003F SLS 1000100\n00003400000003F APP
2023-08-06 03A\n00003500000003F ID   2023-08-06\n00003600000003F DD   2023-08-
06\n00003700000004S IDEX \n00003800000004L ASA T700\n00003900000004L CFAC
DN\n00004000000005S LIST \n00004100000005LONP   (NON-PUB) HENRY,
ARELIA\n00004200000005LOSA   823 W MERRITT PKWY, GILA BEND\n00004600000009S BILL
\n00004700000009L CBR 602 909-3644\n00004800000009L CBE
((ARELIA0@YAHOO.COM))\n00004900000009L OBAN PPB04110168\n00005000000009L MCN

XXXAHXXXXXX\n00005100000009L TRAK HSICX\n00005200000010S S&E \n00005300000010UOGFR
1\n00005400000010F TN   928 683-2049\n00005500000010F NII CO-ETH-A2P\n00005600000010F IPTV
\n00005700000010F ADSL \n00005800000010UOEFX4S1\n00005900000010F TN   928 683-
2049\n00006000000010F IHST USWNETPHNX\n00006100000010F VCID
4\n00006200000010UODXLYH1\n00006300000010F TN   928 683-2049\n00006400000011S RMKS
\n00006500000011L ACC C:05:00P\n00006600000011L RMK ORDER CONSTRUCTED BY
TOM\n00006700000011L RMK LCON 6029093644 ARELIA HENRY\n00006800000011L RMK ***ESHOP
SIMPLEPAY ACCOUNT***\n00006900000011L RMK BILLING ACCOUNT NUMBER\n00007000000011
PPB04110168\n00007100000011L RMK DO NOT MAKE CHANGES CONTACT\n00007200000011
SIMPLE SUPPORT\n00007300000011L RMK PLEASE CONTACT C2E SUPPORT\n00007400000011   800-
853-4633 FOR ANY\n00007500000011   QUESTIONS OR ASSISTANCE WITH\n00007600000011   THIS
ORDER\n"}

Sample output

{'START': [{'sqNo': '000001', 'parent': '000000', 'sectionNo': '00', 'varType': 'S', 'action': '', 'varName': 'STAR
T', 'varValue': 'D33195471'}, {'sqNo': '000002', 'parent': '000000', 'sectionNo': '00', 'varType': 'X', 'action': '',
'varName': 'ORDST', 'varValue': 'AO'}, {'sqNo': '000003', 'parent': '000000', 'sectionNo': '00', 'varType': 'X', '
action': '', 'varName': 'FUNC', 'varValue': 'PRE'}, {'sqNo': '000004', 'parent': '000000', 'sectionNo': '00', 'var
Type': 'X', 'action': '', 'varName': 'STCD', 'varValue': 'AZ'}, {'sqNo': '000005', 'parent': '000000', 'sectionNo': '
00', 'varType': 'X', 'action': '', 'varName': 'ENTOP', 'varValue': 'TOMDMN'}, {'sqNo': '000006', 'parent': '0000
00', 'sectionNo': '00', 'varType': 'X', 'action': '', 'varName': 'ENTDT', 'varValue': '2023-08-06'}, {'sqNo': '0000
07', 'parent': '000000', 'sectionNo': '00', 'varType': 'X', 'action': '', 'varName': 'ENTTM', 'varValue': '03.36.03
'}, {'sqNo': '000008', 'parent': '000000', 'sectionNo': '00', 'varType': 'X', 'action': '', 'varName': 'MODOP', 'var
Value': 'TOMDMN'}, {'sqNo': '000009', 'parent': '000000', 'sectionNo': '00', 'varType': 'X', 'action': '', 'varNa
me': 'MODDT', 'varValue': '2023-08-06'}, {'sqNo': '000010', 'parent': '000000', 'sectionNo': '00', 'varType': 'X
', 'action': '', 'varName': 'MODTM', 'varValue': '03.36.03'}, {'sqNo': '000011', 'parent': '000000', 'sectionNo': '
00', 'varType': 'X', 'action': '', 'varName': 'DISDT', 'varValue': '23218'}, {'sqNo': '000012', 'parent': '000000', '
sectionNo': '00', 'varType': 'X', 'action': '', 'varName': 'DISTM', 'varValue': '0336'}, {'sqNo': '000013', 'parent':
 '000000', 'sectionNo': '00', 'varType': 'X', 'action': '', 'varName': 'SECNT', 'varValue': '0047'}, {'sqNo': '0000
14', 'parent': '000000', 'sectionNo': '00', 'varType': 'X', 'action': '', 'varName': 'WRCEN', 'varValue': '928683'},
 {'sqNo': '000015', 'parent': '000000', 'sectionNo': '00', 'varType': 'X', 'action': '', 'varName': 'ENTCD', 'varVal
ue': 'F'}, {'sqNo': '000016', 'parent': '000000', 'sectionNo': '00', 'varType': 'X', 'action': '', 'varName': 'TYPIN',
'varValue': 'R'}, {'sqNo': '000017', 'parent': '000000', 'sectionNo': '00', 'varType': 'X', 'action': '', 'varName': 'T
YPOT', 'varValue': 'R'}, {'sqNo': '000018', 'parent': '000000', 'sectionNo': '00', 'varType': 'X', 'action': '', 'varN
ame': 'DDCD', 'varValue': 'X'}, {'sqNo': '000019', 'parent': '000000', 'sectionNo': '00', 'varType': 'X', 'action': '
', 'varName': 'OWNRS', 'varValue': '6'}, {'sqNo': '000020', 'parent': '000000', 'sectionNo': '00', 'varType': 'X',

'action': '', 'varName': 'MODAP', 'varValue': '6'}], 'HDR ': [{'sqNo': '000022', 'parent': '000000', 'sectionNo': '
02', 'varType': 'S', 'action': '', 'varName': 'HDR', 'varValue': ''},…….

Observations

- Some rows in the ‘Baseline_Request’ had incomplete JSON format, hence added exception handling code to handle such errors.
- Made an enhancement required where the variable values of lines with variable type ' ' or 'N' get appended to the previous line's value.