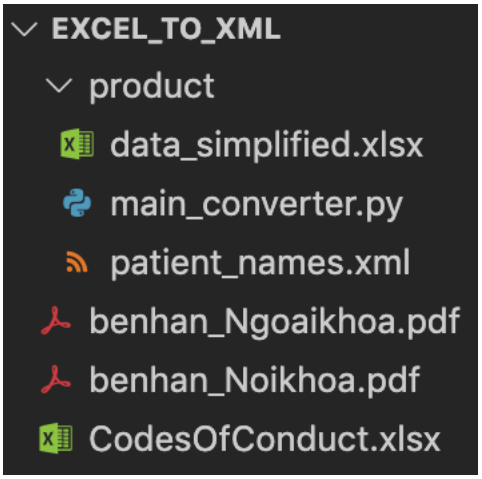
Report Mở Rộng Xử Lý Số Tín Hiệu 23/11/2020

Giáo viên hướng dẫn: Thầy Nguyễn Thanh Tuấn
Sinh viên thực hiện: Thái Quang Nguyên
MSSV: 1813294

***Chủ đề:* Chuyển đổi định dạng .xlsx thành định dạng .xml dùng ngôn ngữ Python.**

I. Cấu tạo thư mục & thư viện cần có:

1. Cấu tạo thư mục:



Hình 1.1. Cấu trúc thư mục

Truy cập repo của project tại link này.
Dựa trên trang web chứa quy định chuẩn và định dạng dữ liệu trong quản lý y tế của *thuvienphapluat.vn*, em đã soạn lại một số tiêu chuẩn trong file excel *CodesOfConduct.xlsx* ở thư mục ngoài *EXCEL_TO_XML.* File này chứa những tiêu chuẩn của các trường của dữ liệu, chia làm 4 sheets, minh hoạ như hình 1.2.

Ở những file trong thư mục con *product*, chứa file *data_simplified.xlsx* là file excel để trích xuất dữ liệu. File *main_converter.py* là file code python để đọc và xuất dữ liệu, data đó cuối cùng được lưu trong file *patient_name.xml*.

Bảng 1. Chỉ tiêu tổng hợp khám bệnh, chữa bệnh BHYT

STT	Chỉ tiêu	Kiểu dữ liệu	Kích thước tối đa	Diễn giải
1	MA_LK	Chuỗi	100	Mã đợt điều trị duy nhất (dùng để liên kết giữa bảng tổng hợp (bảng 1) và các bảng chi tiết (tù
2	STT	Số	10	STT tăng từ 1 đến hết trong 1 lần gửi dữ liệu.
3	MA_BN	Chuỗi	100	Mã số bệnh nhân quy định tại cơ sở khám bệnh, chữa bệnh.
4	HO_TEN	Chuỗi	255	Họ và tên người bệnh
5	NGAY_SINH	Chuỗi	8	Ngày sinh ghi trên thẻ gồm 8 ký tự; 4 ký tự năm + 2 ký tự tháng + 2 ký tự ngày (nếu không có
6	GIOI_TINH	Số	1	Giới tính; Mã hóa (1: Nam; 2: Nữ; 3: Chưa xác định)
7	DIA_CHI	Chuỗi	1024	Ghi địa chỉ theo địa chỉ trên thẻ BHYT hoặc nơi cư trú hiện tại của người bệnh: số nhà (nếu có
8	MA_THE	Chuỗi	n	- Mã thẻ BHYT do cơ quan BHXH cấp - Trường hợp chưa có thẻ BHYT nhưng vẫn được hưởng quyền lợi BHYT, Ví dụ: trẻ em, người ghép tạng,... Ví dụ: TE101KT00000011 (Mã thẻ tạm cho trẻ em thứ 11 đến khám, giấy khai sinh/chứng sinh cấp tại Hà N - Trường hợp trong thời gian điều trị, người bệnh được cấp thẻ BHYT mới có thay đổi thông tin liên quan đ
9	MA_DKBD	Chuỗi	n	Mã cơ sở khám bệnh, chữa bệnh nơi người bệnh đăng ký ban đầu ghi trên thẻ BHYT, gồm có - Trường hợp trong thời gian điều trị, người bệnh được cấp thẻ BHYT mới có thay đổi thông tin - Trường hợp chưa có thẻ BHYT: Ghi mã đơn vị hành chính của tỉnh/TP + 000. Ví dụ: Hà Nội t
10	GT_THE_TU	Chuỗi	n	Thời điểm thẻ có giá trị gồm 8 ký tự; 4 ký tự năm + 2 ký tự tháng + 2 ký tự ngày - Trường hợp trong thời gian điều trị, người bệnh được cấp thẻ BHYT mới có thay đổi thông tin liên quan đ - Trường hợp chưa có thẻ BHYT: Thay thời điểm thẻ có giá trị bằng ngày người bệnh đến khám bệnh, chữa
11	GT.THE_DEN	Chuỗi	n	Thời điểm thẻ hết giá trị gồm 8 ký tự; 4 ký tự năm + 2 ký tự tháng + 2 ký tự ngày - Trường hợp trong thời gian điều trị, người bệnh được cấp thẻ BHYT mới có thay đổi thông tin - Trường hợp chưa có thẻ BHYT: Thay thời điểm thẻ hết giá trị bằng ngày người bệnh ra viện
12	MIEN_CUNG_CT	Chuỗi	8	- Thời điểm người bệnh bắt đầu được hưởng miễn cùng chi trả theo giấy xác nhận của cơ quan BHXH, gồm Ví dụ: ngày 31/03/2017 được hiển thị là: 20170331 - Nếu không có giấy xác nhận miễn cùng chi trả của cơ quan BHXH thì để trống
13	TEN_BENH	Chuỗi	n	Ghi đầy đủ các chẩn đoán được ghi trong hồ sơ, bệnh án

Hình 1.2. Nội dung minh hoạ của file CodeOfConduct.xlsx

2. Các thư viện cần có:

- openpyxl: thư viện dùng để đọc/ghi file Excel theo những định dạng xlsx/xlsm/xltx/xltm.
- yattag: thư viện dùng để tạo ra file HTML hoặc XML bằng code Python.
- datetime: thư viện để đọc và tính toán ngày tháng trong Python.

II. Diễn giải code:

1. Cách thức đọc file excel:

Trước tiên ta import hàm *load_workbook()* từ thư viện *openpyxl* và gọi hàm với đối số là tên của file excel chứa dữ liệu:

```
from openpyxl import load_workbook
wb = load_workbook("data_simplified.xlsx")
ws = wb.worksheets[0]
```

Tiếp đến ta import thư viện yattag để xuất định dạng xml:

```
from yattag import Doc, indent
# Create Yattag doc, tag and text objects
doc, tag, text = Doc().tagtext()
```

Class yattag.Doc hoạt động như cách ta liên kết các chuỗi lại với nhau, ví dụ đơn giản như hình dưới:

```
mylist = []
mylist.append('Everybody')
mylist.append('likes')
mylist.append('pandas.')
mystring = ' '.join(mylist) # mystring contains "Everybody likes pandas."
```

Hình 2.1. Cách thức hoạt động của class yattag.Doc

	A	B	C
1	**Chỉ tiêu**	**Bệnh nhân 1**	**Bệnh nhân 2**
2	MA_LK	LK123122	LK123123
3	STT	1	1
4	MA_BN	BN123123	BN123124
5	HO_TEN	NGUYỄN VĂN NỘI KHOA	NGUYỄN THỊ NGOẠI KHOA
6	NGAY_SINH	1/1/20	1/1/00
7	GIOI_TINH	Nam	Chưa xác định
8	DIA_CHI	ận Ba Đình, Thành phố Hà Nội	, Quận Ba Đình, Thành phố Hà Nội
9	MA_THE	HS1231231222222	HS1231231222222
10	MA_DKBD	DKBD1000	DKBD1001
11	GT_THE_TU	12/3/00	12/3/00
12	GT_THE_DEN	12/3/00	12/3/00
13	MIEN_CUNG_CT	12/4/00	12/5/00
14	TEN_BENH	điên khùng	điên khùng
15	MA_BENH	BENH22222	BENH22223
16	MA_BENHKHAC	BENHKHAC22222	BENHKHAC22223
17	MA_LYDO_VVIEN	Đúng tuyến	Đúng tuyến
18	MA_NOI_CHUYEN	NOICHUYEN12345	NOICHUYEN12346

Hình 2.2. Nội dung file data_simplified.xlsx

Tạo mẫu file *data_simplified.xlsx* chứa nội dung là các trường (cột A) và các thông số tương ứng của từng bệnh nhân (từ cột B trở đi).

Bây giờ ta sẽ đọc lần lượt từng bệnh nhân (từng cột B, C), trong mỗi bệnh nhân ta đọc từng hàng chính là thông số của các trường liên quan đến bệnh nhân. Các dòng code dưới giúp ta làm việc này:

2. Cách thức tạo và validate các trường dữ liệu:

```
with tag('Cac_Benh_Nhan'):
    # Use ws.max_row for all rows
    for col in ws.iter_cols(min_col=2, max_col=3, min_row=2, max_row=41):
        col = [cell.value for cell in col]
```

Tạo một tag *<Cac_Benh_Nhan></Cac_Benh_Nhan>* để lưu trữ thông tin từ file *data_simplified.xlsx* trong một lần đọc dữ liệu. Dữ liệu của chúng ta bắt đầu từ hàng 2 cho đến hàng 41, và từ cột B cho đến C (tức là cột 2 và 3), ta khai báo *min_row*, *max_row*, *min_col*, *max_col* như trên là đối số của hàm *iter_cols()*. Ta lưu tất cả các giá trị của từng cell của một cột vào mảng một chiều *col*, sau này ta có thể truy xuất từng giá trị đó thông qua index của mảng *col*.

```
25          with tag("Benh_Nhan"):
26              with tag("MA_LK", klass='code', type='string_100'):…
32              with tag("STT", klass='number', type='int_10'):…
38              with tag("MA_BN", klass='code', type='string_100'):…
44              with tag("HO_TEN", klass='detail', type='string_255'):…
50              with tag("NGAY_SINH", klass='time', type='year_month_day'):…
57              with tag("GIOI_TINH", klass='detail', type='selection_1'):…
67              with tag("DIA_CHI", klass='detail', type='string_1024'):…
73              with tag("MA_THE", klass='code', type='string_n'):…
79              with tag("MA_DKBD", klass='code', type='string_n'):…
85              with tag("GT_THE_TU", klass='time', type='year_month_day'):…
91              with tag("GT_THE_DEN", klass='time', type='year_month_day'):…
97              with tag("MIEN_CUNG_CT", klass='time', type='year_month_day'):…
103             with tag("TEN_BENH", klass='detail', type='string_n'):…
109             with tag("MA_BENH", klass='code', type='string_15'):…
115             with tag("MA_BENHKHAC", klass='code', type='string_255'):…
121             with tag("MA_LYDO_VVIEN", klass='detail', type='selection_1'):…
133             with tag("MA_NOI_CHUYEN", klass='code', type='string_5'):…
139             with tag("MA_TAI_NAN", klass='code', type='int_1'):…
145             with tag("NGAY_VAO", klass='time', type='year_month_day_hour_minute'):…
151             with tag("NGAY_RA", klass='time', type='year_month_day_hour_minute'):…
157             with tag("SO_NGAY_DTRI", klass='time', type='int_3'):…
163             with tag("KET_QUA_DTRI", klass='detail', type='selection_1'):…
177             with tag("TINH_TRANG_RV", klass='detail', type='selection_1'):…
189             with tag("NGAY_TTOAN", klass='time', type='year_month_day'):…
195             with tag("T_THUOC", klass='money', type='float_15_decimal_2'):…
201             with tag("T_VTYT", klass='money', type='float_15_decimal_2'):…
207             with tag("T_TONGCHI", klass='money', type='float_15_decimal_2'):…
213             with tag("T_BNTT", klass='money', type='float_15_decimal_2'):…
219             with tag("T_BNCCT", klass='money', type='float_15_decimal_2'):…
225             with tag("T_BHTT", klass='money', type='float_15_decimal_2'):…
231             with tag("T_NGUONKHAC", klass='money', type='float_15_decimal_2'):…
237             with tag("T_NGOAIDS", klass='money', type='float_15_decimal_2'):…
243             with tag("NAM_QT", klass='time', type='year'):…
249             with tag("THANG_QT", klass='time', type='month'):…
255             with tag("MA_LOAI_KCB", klass='code', type='selection_1'):…
265             with tag("MA_KHOA", klass='code', type='string_15'):…
271             with tag("MA_CSKCB", klass='code', type='string_5'):…
277             with tag("MA_KHUVUC", klass='code', type='selection_2'):…
287             with tag("MA_PTTT_QT", klass='code', type='string_255'):…
293             with tag("CAN_NANG", klass='detail', type='float_5_decimal_2'):…
```

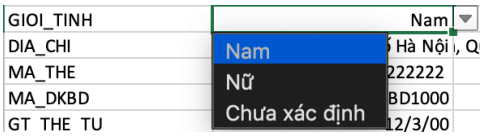
Với mỗi một trường dữ liệu, ta sẽ tạo một tag với tên tương ứng, cú pháp như hình bên. Ta có thể chia các trường dữ liệu theo *class* và *type*, ví dụ như class 'time' để chỉ thời gian và type *'year_month_day'* để chỉ quy định của nội dung tag đó.
Type 'int_5' là số integer khi chuyển thành chuỗi bằng lệnh str() thì len(str()) sẽ bé hơn hoặc bằng 5, tương ứng với yêu cầu trong file *CodeOfConduct.xlsx*.

Hình 2.3. Các tag và class, type tương ứng

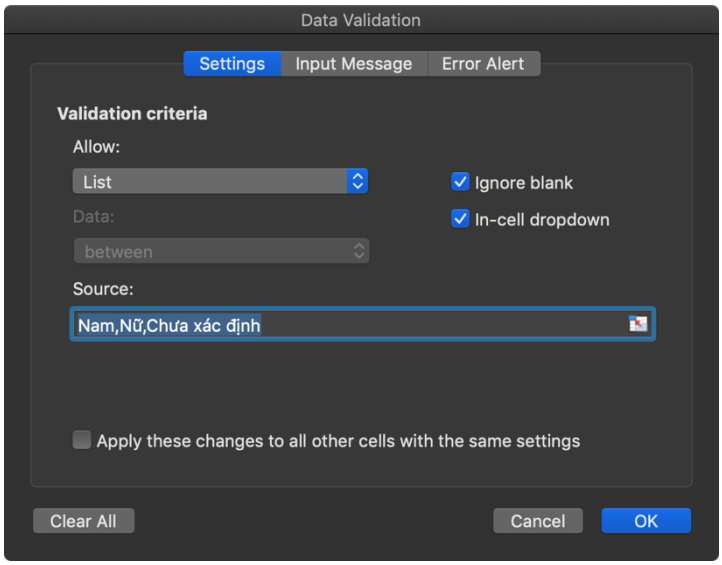
Ở mỗi trường giá trị khi đọc vào, ta check xem có giá trị hay không hàm *if col[count] == None*, biến *count* được dùng để đọc lần lượt các hàng trong một cột, khi đọc xong một hàng thì sẽ được cộng lên 1. Điều kiện tiếp theo chính là chiều dài của dữ liệu có thoả mãn yêu cầu không, ta có ví dụ như trường dữ liệu *MA_LK*:

```
with tag("MA_LK", klass='code', type='string_100'):
                if col[count] == None or len(str(col[count])) > 100:
                    text("NODATA")
                else:
                    text(col[count])
                count += 1
```

Ở những trường như là ‘GIOI_TINH’, giá trị nhập vào excel được phân thành những option là ‘Nam’, ‘Nữ’ hoặc ‘Chưa xác định’. Điều này được thực hiện trong file *data_simplified.xlsx* bằng chức năng Data Validation như sau:



Hình 2.4. Chức năng Data Validation trong EXCEL để tạo option box



Nhờ đó, ta có thể rào được khoảng giá trị nhập vào của người nhập. Tiếp đến ta đọc cell này ở code python như sau:

```
with tag("GIOI_TINH", klass='detail', type='selection_1'):
    if col[count] == None:
        text("NODATA")
    elif col[count] == "Nam":
        text('1')
    elif col[count] == "Nữ":
        text('2')
    elif col[count] == "Chưa xác định":
        text('3')
    count += 1
```

Tương tự như hàm switch case của C, ta xuất ký tự ‘1’, ‘2’, ‘3’ tương tự với các giá trị đầu vào là ‘Nam’, ‘Nữ’ và ‘Chưa xác định’.

Ở trường dữ liệu là thời gian, nhờ vào thư viện datetime, ta check xem đó có phải là biến datetime hay không bằng hàm *type()*, sau đó xuất ra theo định dạng mong muốn là *‘yyyymmdd’* bằng hàm *strftime(“%Y%m%d”)*.

```
with tag("NGAY_SINH", klass='time', type='year_month_day'):
    if type(col[count]) is datetime.datetime:
        temp_date = col[count]
        text(col[count].strftime("%Y%m%d"))
    else:
        text("NODATA")
    count += 1
```

Ở đây, vì ở trường dữ liệu CAN_NANG ở cuối, ta chỉ thu thập dữ liệu đối với các trẻ em dưới 1 tuổi, nên ta sẽ dùng thư viện datetime để lấy giá trị của thời điểm hiện tại, trừ đi cho giá trị

ngày sinh nhập vào xem có bé hơn 365 ngày không (ở đây chưa xét đến yếu tố năm nhuận hay múi giờ).

40	CAN_NANG	số	5	Chỉ thu thập với các bệnh nhân là trẻ em dưới 1 tuổi. Là số kilogam (kg) cân nặng của trẻ em khi vào viện. Biểu thị đầy đủ cả Số thập phân, dấu thập phân là dấu chấm (.), ghi đến 2 chữ số sau dấu thập phân. Ví dụ: 5.75 kg.

Hình 2.5. Quy định của trường dữ liệu CAN_NANG

```
current_date = datetime.datetime.now()
temp_date = datetime.datetime(2020, 11, 22)

#...

with tag("CAN_NANG", klass='detail', type='float_5_decimal_2'):
    if (current_date - temp_date).days <= 365:
        if type(col[count]) == float and len(str(col[count])) <= 5:
            text(col[count])
            age = (current_date - temp_date).days
            print(current_date)
            print(temp_date)
            print(age)
    else:
        text("NODATA")
```

Biến *current_date* được gán cho thời điểm đọc file excel, biến *temp_date* được khởi tạo và về sau được gán bằng giá trị ngày sinh của bệnh nhân. Hiệu của hai giá trị này tính ra số ngày nếu thoả bé hơn hoặc bằng 365 thì thoả mãn, tính được bằng hàm *(current_date - temp_date).days*.

Cuối cùng, ta có thể lưu lại các giá trị gồm các tag, text vào file *patient_names.xml*:

```
result = indent(
    doc.getvalue(),
    indentation = '    '
)

with open("patient_names.xml", "w") as f:
    f.write(result)
```

Kết quả thu được như hình dưới:

main_converter.py | patient_names.xml

product > patient_names.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"></xs:schema>
<Cac_Benh_Nhan>
    <Benh_Nhan>
        <MA_LK class="code" type="string_100">LK123122</MA_LK>
        <STT class="number" type="int_10">1</STT>
        <MA_BN class="code" type="string_100">BN123123</MA_BN>
        <HO_TEN class="detail" type="string_255">NGUYỄN VĂN NỘI KHOA</HO_TEN>
        <NGAY_SINH class="time" type="year_month_day">20200101</NGAY_SINH>
        <GIOI_TINH class="detail" type="selection_1">1</GIOI_TINH>
        <DIA_CHI class="detail" type="string_1024">27 phố Hàng Đậu, phường Trúc Bạch, Quận Ba Đình, Thành phố
        <MA_THE class="code" type="string_n">HS1231231222222 </MA_THE>
        <MA_DKBD class="code" type="string_n">DKBD1000</MA_DKBD>
        <GT_THE_TU class="time" type="year_month_day">20001203</GT_THE_TU>
        <GT_THE_DEN class="time" type="year_month_day">20001203</GT_THE_DEN>
        <MIEN_CUNG_CT class="time" type="year_month_day">20001204</MIEN_CUNG_CT>
        <TEN_BENH class="detail" type="string_n">điên khùng</TEN_BENH>
        <MA_BENH class="code" type="string_15">BENH22222</MA_BENH>
        <MA_BENHKHAC class="code" type="string_255">BENHKHAC22222</MA_BENHKHAC>
        <MA_LYDO_VVIEN class="detail" type="selection_1">1</MA_LYDO_VVIEN>
        <MA_NOI_CHUYEN class="code" type="string_5">NODATA</MA_NOI_CHUYEN>
        <MA_TAI_NAN class="code" type="int_1">1</MA_TAI_NAN>
        <NGAY_VAO class="time" type="year_month_day_hour_minute">200012030000</NGAY_VAO>
        <NGAY_RA class="time" type="year_month_day_hour_minute">200012030000</NGAY_RA>
        <SO_NGAY_DTRI class="time" type="int_3">NODATA</SO_NGAY_DTRI>
        <KET_QUA_DTRI class="detail" type="selection_1">5</KET_QUA_DTRI>
        <TINH_TRANG_RV class="detail" type="selection_1">2</TINH_TRANG_RV>
        <NGAY_TTOAN class="time" type="year_month_day">NODATA</NGAY_TTOAN>
        <T_THUOC class="money" type="float_15_decimal_2">345.679</T_THUOC>
        <T_VTYT class="money" type="float_15_decimal_2">345.679</T_VTYT>
        <T_TONGCHI class="money" type="float_15_decimal_2">345.679</T_TONGCHI>
        <T_BNTT class="money" type="float_15_decimal_2">345.679</T_BNTT>
        <T_BNCCT class="money" type="float_15_decimal_2">345.679</T_BNCCT>
        <T_BHTT class="money" type="float_15_decimal_2">345.679</T_BHTT>
        <T_NGUONKHAC class="money" type="float_15_decimal_2">345.679</T_NGUONKHAC>
        <T_NGOAIDS class="money" type="float_15_decimal_2">345.679</T_NGOAIDS>
        <NAM_QT class="time" type="year">2000</NAM_QT>
        <THANG_QT class="time" type="month">12</THANG_QT>
```

Hình 2.6. File patient_names.xml xuất ra được