**G3.29**

**HỌ TÊN: Nguyễn Diệu Hương MSSV: 21110489 KÝ TÊN: HUONG**

**TRƯỜNG: HCMUTE**

**MÔN: Lập trình Python**

**NGÀY: 12/04/2023 (BUỔI HỌC SỐ 10)**

**Riêng buổi học Phòng máy ghi thêm, SỐ MÁY: PHÒNG MÁY:**

============

**Buổi 10.1\_... = Bài tập 4\_Phần 3: Pre-processing/Xử Thuộc tính NULL (NULL Attributes Pro..)**

**Hỗ trợ Tiền xử lý dữ liệu**

1. **Giới thiệu chủ đề**

+ EXAMPLE2: Hỗ trợ Pre-processing/Xử Thuộc tính NULL (NULL Attributes Pro..)

NẠP & CHỌN CÁC THUỘC TÍNH -> TIỀN XỬ LÝ DỮ LIỆU EDA (B3: XỬ LÝ CỘT NULL)

+ Thư viện chính GUI : Tkinter

1. **FULL CODES (Tham khảo: sv nên cá nhân hóa thông tin bài làm)**

"""Created on Wed Apr 12 13:33:00 2023

@author: 29\_NGUYEN/DIEUHUONG

NẠP & CHỌN CÁC THUỘC TÍNH -> TIỀN XỬ LÝ DỮ LIỆU EDA (B3: XỬ LÝ CỘT NULL)

"""

import tkinter as tk

#Khởi tạo thiết lập đối tượng WINFORM

class App(*tk*.*Tk*):

    def \_\_init\_\_(*master*):

*super*().\_\_init\_\_()

        master.title("BƯỚC 3: XỬ LÝ CỘT NULL = NẠP & CHỌN CÁC CỘT NULL CẦN XỬ LÝ")

        master.geometry("750x450")

        #Thiết lập label

        tk.Label(master, *text*="Các Thuộc tính").place(*x*=15, *y*=15)

        #Thiết lập textbox (khi apply = dùng for nạp các thuộc tính vào thay vì nhập từ textbox )

        master.txtSource = tk.Entry(master, *width*=30)

        # Entry = cho nhập DL vào

        master.txtSource.place(*x*=120, *y*=15)

        master.btnAdd = tk.Button(master,

*text*="Add",

*width*=10,

*command*=master.InsertData)

        master.btnAdd.place(*x*=320, *y*=10)

        master.direction\_var = tk.StringVar(*value*="thuan")

        master.forward\_radio = tk.Radiobutton(master,

*text*="Copy Forward",

*variable*=master.direction\_var,

*value*="thuan")

        master.backward\_radio = tk.Radiobutton(master,

*text*="Copy Backward",

*variable*=master.direction\_var,

*value*="nghich")

        master.forward\_radio.place(*x*=400, *y*=10)

        master.backward\_radio.place(*x*=500, *y*=10)

        master.listbox1 = tk.Listbox(master,

*height*=25,

*width*=40,

*font*="Consolas 8",

*selectmode*=tk.EXTENDED)

        master.listbox1.bind("<Button-3>", master.ShowPopupMenuA)

        master.listbox1.place(*x*=15, *y*=50)

        master.listbox2 = tk.Listbox(master,

*height*=25,

*width*=40,

*font*="Consolas 8",

*selectmode*=tk.EXTENDED)

        master.listbox2.place(*x*=450, *y*=50)

        master.listbox2.bind("<Button-3>", master.ShowPopupMenuB)

        tk.Label(master, *text*="Số lượng").place(*x*=15, *y*=420)

        master.lblSoLuong = tk.Label(master,

*relief*=tk.SUNKEN,

*font*="Times 8",

*borderwidth*=3,

*width*=15,

*height*=1)

        master.lblSoLuong.place(*x*=100, *y*=420)

        master.btn1 = tk.Button(master,

*text*="Copy to left",

*width*=10,

*command*=master.CopyToLeft)

        master.btn2 = tk.Button(master,

*text*="Copy to right",

*width*=10,

*command*=master.CopyToRight)

        master.btn3 = tk.Button(master,

*text*="Move left <<",

*width*=10,

*command*=master.MoveToLeft)

        master.btn4 = tk.Button(master,

*text*="Move right >>",

*width*=10,

*command*=master.MoveToRight)

        master.btn5 = tk.Button(master,

*text*="Delete X",

*width*=10,

*command*=master.Delete)

        master.btn1.place(*x*=320, *y*=70)

        master.btn2.place(*x*=320, *y*=110)

        master.btn3.place(*x*=320, *y*=150)

        master.btn4.place(*x*=320, *y*=190)

        master.btn5.place(*x*=320, *y*=230)

    #Hàm xử lý btnAdd

    def InsertData(*master*):

        dem = 0

        a = master.txtSource.get().strip()

        kq = a.ljust(20)

        # kq = chuỗi <-từ- chuỗi a = canh chỉnh left & bên right là fillchar(space) -> chiều dài width

        if (a != ""):

            master.listbox1.insert(tk.END, kq)

        #Đếm số dòng trong listbox

        dem = master.listbox1.size()

        #Điền thông tin vào label

        master.lblSoLuong.configure(*text*=dem)

        master.txtSource.delete(0, tk.END)# xóa trống text -> để nhập mới

    #Hàm CopyToRight từ listbox1 sang listbox2 (cho phép chọn nhiều)= có thể viết theo cách khác (như bên dưới)

    def CopyToRight(*master*):

        direction = master.direction\_var.get()

        if direction == "thuan":

            selection = master.listbox1.curselection()

            selection = sorted(selection)  # sắp xếp theo thứ tự tăng dần

            master.listbox2.insert(tk.END,

                                 \*[master.listbox1.get(i) for i in selection])

        else:

            selection = master.listbox1.curselection()

            selection = sorted(selection,

*reverse*=True)  # sắp xếp theo thứ tự giảm dần

            master.listbox2.insert(0, \*[master.listbox1.get(i) for i in selection])

        master.listbox1.select\_clear(0, tk.END)  # b

        master.InsertData()

    #Hàm MoveToRight: di chuyên các thuộc tính từ listbox1 sang listbox2 (xóa các thuộc tính đã chuyển listbox1->)

    def MoveToRight(*master*):

        direction = master.direction\_var.get()# lấy ds các vị trí chọn

        selection = master.listbox1.curselection()

        selection = sorted(selection, *reverse*=(direction == "nghich"))

        if selection:

            values = [master.listbox1.get(i) for i in selection]

            master.listbox2.insert(tk.END, \*values)

        if direction == "thuan":

            for i in reversed(selection):

                master.listbox1.delete(i)

        else:

            for i in selection:

                master.listbox1.delete(i)

        master.listbox1.selection\_clear(0, tk.END)

        master.InsertData()

    def Delete(*master*):

        selection = master.listbox1.curselection()

        if len(selection) > 0:

            for i in reversed(selection):# duyệt ngược , sv tự làm duyệt thuận

                master.listbox1.delete(i)

        selection2 = master.listbox2.curselection()

        if len(selection2) > 0:

            for i in reversed(selection2):

                master.listbox2.delete(i)

        master.InsertData()

    def CopyToLeft(*master*):

        direction = master.direction\_var.get()

        if direction == "thuan":

            selection = master.listbox2.curselection()

            selection = sorted(selection)

            if len(selection) > 0:

                values = [master.listbox2.get(i) for i in selection]

                master.listbox1.insert(tk.END, \*values)

        else:

            selection = master.listbox2.curselection()

            selection = sorted(selection, *reverse*=True)

            if len(selection) > 0:

                values = [master.listbox2.get(i) for i in selection]

                master.listbox1.insert(tk.END, \*values)

        master.InsertData()

    def MoveToLeft(*master*):

        direction = master.direction\_var.get()

        selection = master.listbox2.curselection()

        selection = sorted(selection, *reverse*=(direction == "nghich"))

        if selection:

            values = [master.listbox2.get(i) for i in selection]

            master.listbox1.insert(tk.END, \*values)

            if direction == "thuan":

                for i in reversed(selection):

                    master.listbox2.delete(i)

            else:

                for i in selection:

                    master.listbox2.delete(i)

        master.listbox2.selection\_clear(0, tk.END)

        master.InsertData()

    def ShowPopupMenuA(*master*, *e*):

        if master.listbox1.size() > 0:

            popMenu = tk.Menu(master.listbox1, *tearoff*=tk.FALSE)

            popMenu.add\_command(*command*=master.CopyToRight)

            popMenu.add\_command(*command*=master.MoveToRight)

            popMenu.add\_command(*command*=master.Delete)

            popMenu.tk\_popup(

                e.x\_root,

                e.y\_root)  #phải thiết lập x\_root, y\_root để showpopup

    def ShowPopupMenuB(*master*, *e*):

        if master.listbox1.size() > 0:

            popMenu = tk.Menu(master.listbox2, *tearoff*=tk.FALSE)

            popMenu.add\_command(*command*=master.CopyToLeft)

            popMenu.add\_command(*command*=master.MoveToLeft)

            popMenu.add\_command(*command*=master.Delete)

            popMenu.tk\_popup(

                e.x\_root,

                e.y\_root)  #phải thiết lập x\_root, y\_root để showpopup

app = App()

app.mainloop()

yc1: viết lại các hàm copy, move, delete theo 2 cách: 1. Dùng vòng lặp (vd: while) thông thường 2. Sử dụng phương thức curselection() có sẵn trong listbox

Yc2: Bổ sung vào form: Tương ứng copy/move: checkbox or radioButton: xác định thứ tự copy/move