



UNIVERSIDAD DE PIURA

Curso Python Nivel 1

Trabajo Final

Alcalde Paredes, Carla Valeria
Calopiña Arbulú, Bruno Milton
Fernández Aquino, Henry Ismael Junior
Luna Otero, Vincenzo Antonio
Oquelis Gonzales, Alessandra
Oquelis Guerrero, María José

23 de enero de 2022

Índice

1. Introducción	3
2. Análisis del sistema	4
3. Ejemplos al correr el programa	9
4. Conclusiones.....	13

1. Introducción

En el presente trabajo se han aplicado los conocimientos básicos y herramientas aprendidas en el curso de Python 1, con la finalidad que los alumnos practiquen dichos comandos y teoría aprendida, viéndolos en situaciones que se pueden dar en la vida real.

En base a esto, el objetivo principal del trabajo es ordenar un grupo de datos ilimitado para la situación creada, de tal forma que permita identificar las características del paciente y que se facilite el proceso de prueba COVID. Este se centra en la recolección, organización y análisis de información de las personas que buscan realizarse una prueba, permitiendo que el usuario que lo está ejecutando ingrese su información personal de forma ordenada y posteriormente conteste algunas preguntas.

De acuerdo a la respuesta de dichas preguntas, se irán clasificando los datos para que puedan contabilizarse, y así hacer gráficos que permitan analizar las respuestas de los usuarios.

Además, el programa permitirá que se agreguen nuevas filas; sin embargo, no se agregarán nuevas columnas ya que estas están ya estipuladas.

Asimismo, toda la información dada por los usuarios se guardará en un archivo Excel, de tal forma que este se actualice cada vez que ingresa un nuevo usuario, sin borrar los anteriores.

2. Análisis del sistema

```
import pandas as pd
from pandas import DataFrame

#Se escribe pd para no tener que escribir pandas a lo largo del código. De pandas se importa DataFrame para elaborar tablas.

lista_encabezados_0 = {'Nombres':[], 'Apellidos':[], 'Edad':[], 'Sexo':[], 'Tipo de documento de identidad':[],
                      'Número del documento':[], 'Dirección':[], 'Número celular':[], 'Correo':[], 'Ocupación':[]}

#Se utiliza un diccionario donde cada apartado será la división de una nueva columna.
#Los corchetes [] son necesarios para que se guarden los datos en una lista de datos. Siempre se agregarán datos en columnas.
tabla_datos_personales = pd.DataFrame(lista_encabezados_0)
tabla_datos_personales_df = tabla_datos_personales

#pd.DataFrame es para que el diccionario ya obtenido se pueda organizar en una tabla de datos. Luego se le pone otro nombre en la siguiente línea.

##Se repite el procedimiento en las siguientes 3 listas de encabezados

lista_encabezados = {'Tos':[], 'Dolor de garganta':[], 'Congestión nasal':[], 'Dificultad respiratoria':[],
                    'Fiebre':[], 'Diarrea':[], 'Náuseas/vómitos':[], 'Dolor muscular':[], 'Dolor de pecho':[]}
tabla_sintomas = pd.DataFrame(lista_encabezados)
tabla_sintomas_df = tabla_sintomas

lista_encabezados_1 = {'Mayor de 65':[], 'Enfermedad cardiovascular':[], 'Diabetes':[], 'Enfermedad cerebrovascular':[],
                      'Síndrome de Down':[], 'Obesidad':[], 'Embarazo':[], 'Inmunodeficiencia':[], 'Enfermedad renal':[],
                      'Enfermedad pulmonar':[], 'Cáncer':[], 'Receptor de órganos':[]}
tabla_riesgos = pd.DataFrame(lista_encabezados_1)
tabla_riesgos_df = tabla_riesgos
```

```
lista_encabezados_2 = {'¿Los últimos 7 días ha tenido contacto con alguien con Covid?':[], '¿Se encuentra vacunado?':[],
                      '¿Cuántas dosis?':[], '¿Ha presentado Covid?':[], '¿Qué tipo de prueba desea?':[]}
tabla_covid = pd.DataFrame(lista_encabezados_2)
tabla_covid_df = tabla_covid

#A continuación se definen 4 funciones, las cuales correrán cuando sean llamadas

def añadir_fila_datos_personales():
    global tabla_datos_personales_df
    dicc_datos_personales = {}
    columnas_2_df = lista_encabezados_0
    for columna_2 in columnas_2_df:
        #La columna_2 va a tener todos los valores de columnas_2_df
        dicc_datos_personales[columna_2] = input('Ingrese {}: '.format(columna_2))
    tabla_datos_personales_df = tabla_datos_personales_df.append(dicc_datos_personales, ignore_index = True)
#Todos los datos de la lista_encabezados_0 serán ingresados y guardados acá por ser igual a columnas_2_df
```

```
##Se repite el mismo proceso con las siguientes funciones en este apartado##

def añadir_fila_sintomas():
    global tabla_sintomas_df
    dicc_sintomas = {}
    columnas_df = lista_encabezados
    for columna in columnas_df:
        #columna va a tener todos los valores de columnas_df
        dicc_sintomas[columna] = input('¿Tiene o ha tenido {}: '.format(columna))
    tabla_sintomas_df = tabla_sintomas_df.append(dicc_sintomas, ignore_index = True)
#Todos los datos de la lista_encabezados serán ingresados y guardados acá por ser igual a columnas_df

def añadir_fila_riesgos():
    global tabla_riesgos_df
    dicc_riesgos = {}
    columnas_1_df = lista_encabezados_1
    for columna_1 in columnas_1_df:
        #La columna_1 va a tener todos los valores de columnas_1_df
        dicc_riesgos[columna_1] = input('¿Pertenece al siguiente grupo de riesgo: {}: '.format(columna_1))
    tabla_riesgos_df = tabla_riesgos_df.append(dicc_riesgos, ignore_index = True)
#Todos los datos de la lista_encabezados_1 serán ingresados y guardados acá por ser igual a columnas_1_df
```

```
def añadir_fila_covid():
    global tabla_covid_df
    dicc_covid = {}
    columnas_0_df = lista_encabezados_2
    for columna_0 in columnas_0_df:
        #La columna_0 va a tener todos los valores de columnas_0_df
        dicc_covid[columna_0] = input('{}: '.format(columna_0))
    tabla_covid_df = tabla_covid_df.append(dicc_covid, ignore_index = True)
#Todos los datos de la lista_encabezados_2 serán ingresados y guardados acá por ser igual a columnas_0_df
```

```
def Uniónfiltrar():
    global tabla_unión
    global tabla_unión_1
    tabla_unión = tabla_datos_personales_df.join(tabla_sintomas_df)
    #Se utiliza el comando df.join para unir tabla_datos_personales_df con tabla_sintomas_df
    tabla_unión_1 = tabla_datos_personales_df.join(tabla_riesgos_df)
    #Se utiliza el comando df.join para unir tabla_datos_personales_df con tabla_riesgos_df
    encabezado = input('Ingrese el valor del encabezado a filtrar: ')
    filtro = input('Ingrese el valor del filtro: ')
    respuesta = tabla_unión.loc[tabla_unión[encabezado] == filtro]
    #Se utiliza el comando df.loc para filtrar según etiquetas, dentro de tabla_unión, y se guarda en la variable respuesta
    respuesta_1 = tabla_unión_1.loc[tabla_unión_1[encabezado] == filtro]
    #Se utiliza el comando df.loc para filtrar según etiquetas, dentro de tabla_unión_1, y se guarda en la variable respuesta_1
    display(respuesta)
    #Se muestra el dataframe respuesta
    display(respuesta_1)
    #Se muestra el dataframe respuesta_1

def agregar_paciente_0():
    decisión_0 = input("Escriba 'SI' si desea ingresar a un paciente a la base datos, caso contrario, escriba 'NO': ")
    if decisión_0 == 'SI':
        respuesta = True
    else:
        respuesta = False
    return respuesta
#Return devolverá la respuesta True o False al correr la función.
#Si se escribe SI el código correrá la función decision_0=agregar_paciente_0() donde se añadirán filas de datos personales.
```

```
def grafico_barras():
    decisión_1 = input("SI desea ver un gráfico de barras de las personas vs síntomas escriba 'SI', caso contrario, escriba 'NO': ")
    if decisión_1 == 'SI':
        respuesta = True
    else:
        respuesta = False
    return respuesta
#Return devolverá la respuesta True o False al correr la función.
#Si se escribe SI el código correrá la función decision_1=grafico_barras(), lo que mostrará un gráfico de barras de los pacientes vs sus síntomas

def grafico_barras_2():
    decisión_2 = input("SI desea ver un gráfico de barras de las personas vs grupos de riesgo escriba 'SI', caso contrario, escriba 'NO': ")
    if decisión_2 == 'SI':
        respuesta = True
    else:
        respuesta = False
    return respuesta
#Return devolverá la respuesta True o False al correr la función.
#Si se escribe SI el código correrá la función decision_2=grafico_barras_2(), lo que mostrará un gráfico de barras de los pacientes vs sus grupos de riesgo
```

```
def filtrar():
    decisión_3 = input('Si desea filtrar datos según cierto criterio escriba SI, caso contrario, escriba NO: ')
    if decisión_3 == 'SI':
        respuesta = True
    else:
        respuesta = False
    return respuesta
#Return devolverá la respuesta True o False al correr la función.
#Si se escribe SI el código correrá la función decisión_3 = filtrar(), lo cual permitirá filtrar los datos según etiquetas.

import matplotlib.pyplot as plt
from matplotlib.pyplot import subplots

#Se escribe plt para no tener que escribir matplotlib.pyplot a lo largo del código. De matplotlib.pyplot se importa subplots para decidir nombres y tamaños de las

#Se define la función BarrasRiesgos
def BarrasRiesgos():
    mayor_65 = list(tabla_riesgos_df.iloc[0:,0])
    #Se elabora una lista con los datos de tabla_riesgos_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 0
    cant_mayor_65 = mayor_65.count('SI')
    #.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a mayores de 65 años
```

```
##Se realiza el mismo proceso con los siguientes riesgos

    enfermedad_cardiovascular = list(tabla_riesgos_df.iloc[0:,1])
    #Se elabora una lista con los datos de tabla_riesgos_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 1
    cant_enfermedad_cardiovascular = enfermedad_cardiovascular.count('SI')
    #.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a enfermedad cardiovascular

    diabetes = list(tabla_riesgos_df.iloc[0:,2])
    #Se elabora una lista con los datos de tabla_riesgos_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 2
    cant_diabetes = diabetes.count('SI')
    #.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a diabetes

    enfermedad_cerebrovascular = list(tabla_riesgos_df.iloc[0:,3])
    #Se elabora una lista con los datos de tabla_riesgos_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 3
    cant_enfermedad_cerebrovascular = enfermedad_cerebrovascular.count('SI')
    #.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a enfermedad cerebrovascular

    sindrome_down = list(tabla_riesgos_df.iloc[0:,4])
    #Se elabora una lista con los datos de tabla_riesgos_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 4
    cant_sindrome_down = sindrome_down.count('SI')
    #.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a síndrome de Down
```



```

obesidad = list(tabla_riesgos_df.iloc[0:,5])
#Se elabora una lista con los datos de tabla_riesgos_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 5
cant_obesidad = obesidad.count('SI')
#.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a obesidad

embarazo = list(tabla_riesgos_df.iloc[0:,6])
#Se elabora una lista con los datos de tabla_riesgos_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 6
cant_embarazo = embarazo.count('SI')
#.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a embarazo

inmunodeficiencia = list(tabla_riesgos_df.iloc[0:,7])
#Se elabora una lista con los datos de tabla_riesgos_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 7
cant_inmunodeficiencia = inmunodeficiencia.count('SI')
#.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a inmunodeficiencia

enfermedad_renal = list(tabla_riesgos_df.iloc[0:,8])
#Se elabora una lista con los datos de tabla_riesgos_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 8
cant_enfermedad_renal = enfermedad_renal.count('SI')
#.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a enfermedad renal

enfermedad_pulmonar = list(tabla_riesgos_df.iloc[0:,9])
#Se elabora una lista con los datos de tabla_riesgos_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 9
cant_enfermedad_pulmonar = enfermedad_pulmonar.count('SI')
#.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a enfermedad pulmonar

```

```

cáncer = list(tabla_riesgos_df.iloc[0:,10])
#Se elabora una lista con los datos de tabla_riesgos_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 10
cant_cáncer = cáncer.count('SI')
#.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a cáncer

receptor_organos = list(tabla_riesgos_df.iloc[0:,11])
#Se elabora una lista con los datos de tabla_riesgos_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 11
cant_receptor_organos = receptor_organos.count('SI')
#.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a receptor de órganos

lista_cant = [cant_mayor_65,cant_enfermedad_cardiovascular,cant_diabetes,cant_enfermedad_cerebrovascular,
cant_sindrome_down,cant_obesidad,cant_embarazo,cant_inmunodeficiencia,cant_enfermedad_renal,
cant_enfermedad_pulmonar,cant_cáncer,cant_receptor_organos]
#Se elabora una lista con las cantidades de personas correspondientes a cada factor de riesgo
lista_riesgos = ['Mayor de 65','Enfermedad cardiovascular','Diabetes','Enfermedad cerebrovascular',
'Síndrome de Down','Obesidad','Embarazo','Inmunodeficiencia','Enfermedad renal',
'Enfermedad pulmonar','Cáncer','Receptor de órganos']
#Se elabora una lista con los encabezados de los factores de riesgo
colores = ['blue','red','green','pink','brown','orange','yellow','skyblue','black','violet','purple','silver']
#Se escogen tantos colores como riesgos (12), para elaborar el gráfico de barras

```

```

fig,ax = subplots(figsize = (28,14))
#Con figsize, se decide el tamaño del gráfico a elaborar, en este caso "28" de ancho y "14" de alto
plt.bar(lista_riesgos,lista_cant, color = colores, width = 0.4)
#Con .bar se elabora un gráfico de barras con lista_riesgos en el eje X, lista_cant en el eje Y, los colores seleccionados y ancho de barra de 0.4
plt.title('Pacientes en grupos de riesgo', size = 20)
#Con .title se escoge el nombre del gráfico y tamaño de letra 20
plt.xlabel('Grupos de riesgo',size = 14)
#Con .xlabel se escoge el nombre del eje X y tamaño de letra 14
plt.ylabel('Cantidad',size = 14)
#Con .ylabel se escoge el nombre del eje Y y tamaño de letra 14
plt.show()
#Con.show() se muestra el gráfico

import matplotlib.pyplot as plt
from matplotlib.pyplot import subplots

#Se escribe plt para no tener que escribir matplotlib.pyplot a lo largo del código. De matplotlib.pyplot se importa subplots para decidir nombres y tamaños de las gráficas

#Se define la función BarrasSíntomas
def BarrasSíntomas():
    tos = list(tabla_sintomas_df.iloc[0:,0])
    #Se elabora una lista con los datos de tabla_sintomas_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 0
    cant_tos = tos.count('SI')
    #.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a tos

```

```

##Se realiza el mismo proceso con los siguientes síntomas

dolor_garganta = list(tabla_sintomas_df.iloc[0:,1])
#Se elabora una lista con los datos de tabla_sintomas_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 1
cant_dolor_garganta = dolor_garganta.count('SI')
#.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a dolor de garganta

congestión_nasal = list(tabla_sintomas_df.iloc[0:,2])
#Se elabora una lista con los datos de tabla_sintomas_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 2
cant_congestión_nasal = congestión_nasal.count('SI')
#.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a congestión nasal

dificultad_respiratoria = list(tabla_sintomas_df.iloc[0:,3])
#Se elabora una lista con los datos de tabla_sintomas_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 3
cant_dificultad_respiratoria = dificultad_respiratoria.count('SI')
#.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a dificultad respiratoria

fiebre = list(tabla_sintomas_df.iloc[0:,4])
#Se elabora una lista con los datos de tabla_sintomas_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 4
cant_fiebre = fiebre.count('SI')
#.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a fiebre

diarrea = list(tabla_sintomas_df.iloc[0:,5])
#Se elabora una lista con los datos de tabla_sintomas_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 5
cant_diarrea = diarrea.count('SI')
#.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a diarrea

```

```

nauseas_vómitos = list(tabla_sintomas_df.iloc[0:,6])
#Se elabora una lista con los datos de tabla_sintomas_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 6
cant_nauseas_vómitos = nauseas_vómitos.count('SI')
#.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a náuseas y/o vómitos

dolor_muscular = list(tabla_sintomas_df.iloc[0:,7])
#Se elabora una lista con los datos de tabla_sintomas_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 7
cant_dolor_muscular = dolor_muscular.count('SI')
#.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a dolor muscular

dolor_pecho = list(tabla_sintomas_df.iloc[0:,8])
#Se elabora una lista con los datos de tabla_sintomas_df, df.iloc permite seleccionar los datos a leer, en este caso, filas desde la 0 al infinito y la columna 8
cant_dolor_pecho = dolor_pecho.count('SI')
#.count cuenta la cantidad de "SI" almacenados en la lista anterior, correspondiente a dolor de pecho

```

```

lista_cant_1 = [cant_tos,cant_dolor_garganta,cant_congestión_nasal,cant_dificultad_respiratoria,
cant_fiebre,cant_diarrea,cant_nauseas_vómitos,cant_dolor_muscular,cant_dolor_pecho]
#Se elabora una lista con las cantidades de personas que han presentado cada síntoma
lista_sintomas = ['Tos','Dolor de garganta','Congestión nasal','Dificultad respiratoria',
'Fiebre','Diarrea','Náuseas/vómitos','Dolor muscular','Dolor de pecho']
#Se elabora una lista con los encabezados de los síntomas
colores_1 = ['blue','red','green','pink','brown','orange','yellow','skyblue','black']
#Se escogen tantos colores como síntomas (8)), para elaborar el gráfico de barras

fig,ax_1 = subplots(figsize = (24,14))
#Con figsize, se decide el tamaño del gráfico a elaborar, en este caso "24" de ancho y "14" de alto
plt.bar(lista_sintomas,lista_cant_1, color = colores_1, width = 0.4)
#Con .bar se elabora un gráfico de barras con lista_sintomas en el eje X, lista_cant_1 en el eje Y, los colores seleccionados y ancho de barra de 0.4
plt.title('Pacientes con síntomas', size = 20)
#Con .title se escoge el nombre del gráfico y tamaño de letra 20
plt.xlabel('Síntomas',size = 14)
#Con .xlabel se escoge el nombre del eje X y tamaño de letra 14
plt.ylabel('Cantidad',size = 14)
#Con .ylabel se escoge el nombre del eje Y y tamaño de letra 14
plt.show()
#Con.show() se muestra el gráfico

```

```

#Se elabora la función principal
def main():
    decisión_0 = agregar_paciente_0()
    while decisión_0 == True:
        añadir_fila_datos_personales()
#Después de llenar los datos personales de la función anterior se presentan las siguientes preguntas, que añadirán una nueva columna de datos con la respuesta del usuario.
    print("A continuación deberá escribir 'SI' si ha presentado el síntoma, caso contrario, 'NO' ")
    añadir_fila_sintomas()
#Al escribir añadir_fila_sintomas, se llama a la función correspondiente creada previamente
    print("A continuación deberá escribir 'SI' si pertenece al grupo de riesgo, caso contrario, 'NO' ")
    añadir_fila_riesgos()
#Al escribir añadir_fila_riesgos, se llama a la función correspondiente creada previamente
    print("Para culminar, responda las siguientes preguntas:")
    añadir_fila_covid()
#Al escribir añadir_fila_covid, se llama a la función correspondiente creada previamente
    decisión_0 = agregar_paciente_0()
#Se pregunta si se desea ingresar datos de otro paciente
    else:
        print('Necesita agregar los datos personales para continuar con el registro')
#De no desear agregar datos, aparecerá el mensaje de else

```

```

xlwriter = pd.ExcelWriter('Pacientes Covid 2.xlsx')
#Con pd.ExcelWriter, se crea un documento excel para almacenar los datos, permite guardar las tablas en diferentes hojas de un excel.
tabla_datos_personales_df.to_excel(xlwriter,sheet_name = 'Datos personales', index = False)
tabla_sintomas_df.to_excel(xlwriter,sheet_name = 'Síntomas', index = False)
tabla_riesgos_df.to_excel(xlwriter,sheet_name = 'Riesgos', index = False)
tabla_covid_df.to_excel(xlwriter,sheet_name = 'Preguntas Covid', index = False)
#Con .to_excel, se exportan los datos a un excel, con sheet_name se pueden elaborar diferentes hojas, con index = False, se evita que se sobrescriban.

xlwriter.close()
#Se termina de escribir el documento excel

decisión_1 = grafico_barras()
if decisión_1 == True:
    BarrasSíntomas()
#Al escribir BarrasSíntomas, se llama a la función correspondiente creada previamente
else:
    print('No se desea ver el gráfico de barras')
#De no desear ver la gráfica de Síntomas, aparecerá el mensaje de else

decisión_2 = grafico_barras_2()
if decisión_2 == True:
    BarrasRiesgos()
#Al escribir BarrasRiesgos, se llama a la función correspondiente creada previamente
else:
    print('No se desea ver el gráfico de barras')
#De no desear ver la gráfica de Riesgos, aparecerá el mensaje de else

##Las nuevas columnas añadidas se basan en las funciones escritas previamente en el código. A medida que el usuario responde se agrega nueva información a las diferentes filas.

```

```
decisión_3 = filtrar()
while decisión_3 == True:
    UniónFiltrar()
#Al escribir UniónFiltrar, se llama a la función correspondiente creada previamente.
    decisión_3 = filtrar()
#Se pregunta si se desea filtrar otra vez
    else:
        print('No se ha filtrado')
#De no desear filtrar aparecerá el mensaje de else.

main()
#Se corre la función principal
```


3. Ejemplos al correr el programa

```
Escriba 'SI' si desea ingresar a un paciente a la base datos, caso contrario, escriba 'NO': SI
Ingrese Nombres: Bruno
Ingrese Apellidos: Calopiña Arbulú
Ingrese Edad: 19
Ingrese Sexo: M
Ingrese Tipo de documento de identidad: DNI
Ingrese Número del documento: 74159607
Ingrese Dirección: Pasaje los Sauces 120
Ingrese Número celular: 986220580
Ingrese Correo: brunocalopi@gmail.com
Ingrese Ocupación: Universitario
A continuación deberá escribir 'SI' si ha presentado el síntoma, caso contrario, 'NO'
¿Tiene o ha tenido Tos?: SI
¿Tiene o ha tenido Dolor de garganta?: SI
¿Tiene o ha tenido Congestión nasal?: SI
¿Tiene o ha tenido Dificultad respiratoria?: SI
¿Tiene o ha tenido Fiebre?: SI
¿Tiene o ha tenido Diarrea?: SI
¿Tiene o ha tenido Náuseas/vómitos?: NO
¿Tiene o ha tenido Dolor muscular?: SI
¿Tiene o ha tenido Dolor de pecho?: SI
A continuación deberá escribir 'SI' si pertenece al grupo de riesgo, caso contrario, 'NO'
¿Pertenece al siguiente grupo de riesgo: Mayor de 65?: SI
¿Pertenece al siguiente grupo de riesgo: Enfermedad cardiovascular?: SI
¿Pertenece al siguiente grupo de riesgo: Diabetes?: SI
¿Pertenece al siguiente grupo de riesgo: Enfermedad cerebrovascular?: SI
¿Pertenece al siguiente grupo de riesgo: Síndrome de Down?: SI
```

```
¿Pertenece al siguiente grupo de riesgo: Obesidad?: SI
¿Pertenece al siguiente grupo de riesgo: Embarazo?: SI
¿Pertenece al siguiente grupo de riesgo: Inmunodeficiencia?: NO
¿Pertenece al siguiente grupo de riesgo: Enfermedad renal?: NO
¿Pertenece al siguiente grupo de riesgo: Enfermedad pulmonar?: SI
¿Pertenece al siguiente grupo de riesgo: Cáncer?: NO
¿Pertenece al siguiente grupo de riesgo: Receptor de órganos?: NO
Para culminar, responda las siguientes preguntas:
¿Los últimos 7 días ha tenido contacto con alguien con Covid?: NO
¿Se encuentra vacunado?: SI
¿Cuántas dosis?: 2
¿Ha presentado Covid?: NO
¿Qué tipo de prueba desea?: PCR
Escriba 'SI' si desea ingresar a un paciente a la base datos, caso contrario, escriba 'NO': SI
Ingrese Nombres: Carla
Ingrese Apellidos: Alcalde Paredes
Ingrese Edad: 20
Ingrese Sexo: F
Ingrese Tipo de documento de identidad: DNI
Ingrese Número del documento: 71314505
Ingrese Dirección: Jirón 5 esquinas 545
Ingrese Número celular: 944223038
Ingrese Correo: carlavr.ap@gmail.com
Ingrese Ocupación: Universitario
```

```
A continuación deberá escribir 'SI' si ha presentado el síntoma, caso contrario, 'NO'
¿Tiene o ha tenido Tos?: SI
¿Tiene o ha tenido Dolor de garganta?: SI
¿Tiene o ha tenido Congestión nasal?: SI
¿Tiene o ha tenido Dificultad respiratoria?: SI
¿Tiene o ha tenido Fiebre?: SI
¿Tiene o ha tenido Diarrea?: SI
¿Tiene o ha tenido Náuseas/vómitos?: SI
¿Tiene o ha tenido Dolor muscular?: SI
¿Tiene o ha tenido Dolor de pecho?: SI
A continuación deberá escribir 'SI' si pertenece al grupo de riesgo, caso contrario, 'NO'
¿Pertenece al siguiente grupo de riesgo: Mayor de 65?: SI
¿Pertenece al siguiente grupo de riesgo: Enfermedad cardiovascular?: SI
¿Pertenece al siguiente grupo de riesgo: Diabetes?: SI
¿Pertenece al siguiente grupo de riesgo: Enfermedad cerebrovascular?: SI
¿Pertenece al siguiente grupo de riesgo: Síndrome de Down?: SI
¿Pertenece al siguiente grupo de riesgo: Obesidad?: SI
¿Pertenece al siguiente grupo de riesgo: Embarazo?: SI
¿Pertenece al siguiente grupo de riesgo: Inmunodeficiencia?: SI
¿Pertenece al siguiente grupo de riesgo: Enfermedad renal?: SI
¿Pertenece al siguiente grupo de riesgo: Enfermedad pulmonar?: SI
¿Pertenece al siguiente grupo de riesgo: Cáncer?: SI
¿Pertenece al siguiente grupo de riesgo: Receptor de órganos?: SI
```

Para culminar, responda las siguientes preguntas:
¿Los últimos 7 días ha tenido contacto con alguien con Covid?: NO
¿Se encuentra vacunado?: SI
¿Cuántas dosis?: 2
¿Ha presentado Covid?: NO
¿Qué tipo de prueba desea?: PCR
Escriba 'SI' si desea ingresar a un paciente a la base datos, caso contrario, escriba 'NO': SI
Ingrese Nombres: María José
Ingrese Apellidos: Oquellis Guerrero
Ingrese Edad: 20
Ingrese Sexo: F
Ingrese Tipo de documento de identidad: DNI
Ingrese Número del documento: 70471260
Ingrese Dirección: Las Fresas A1-9
Ingrese Número celular: 964542995
Ingrese Correo: mj.oquellisguerrero@gmail.com
Ingrese Ocupación: Universitario
A continuación deberá escribir 'SI' si ha presentado el síntoma, caso contrario, 'NO'
¿Tiene o ha tenido Tos?: SI
¿Tiene o ha tenido Dolor de garganta?: SI
¿Tiene o ha tenido Congestión nasal?: SI
¿Tiene o ha tenido Dificultad respiratoria?: SI
¿Tiene o ha tenido Fiebre?: SI
¿Tiene o ha tenido Diarrea?: SI
¿Tiene o ha tenido Náuseas/vómitos?: SI
¿Tiene o ha tenido Dolor muscular?: SI
¿Tiene o ha tenido Dolor de pecho?: SI

A continuación deberá escribir 'SI' si pertenece al grupo de riesgo, caso contrario, 'NO'
¿Pertenece al siguiente grupo de riesgo: Mayor de 65?: SI
¿Pertenece al siguiente grupo de riesgo: Enfermedad cardiovascular?: SI
¿Pertenece al siguiente grupo de riesgo: Diabetes?: SI
¿Pertenece al siguiente grupo de riesgo: Enfermedad cerebrovascular?: SI
¿Pertenece al siguiente grupo de riesgo: Síndrome de Down?: SI
¿Pertenece al siguiente grupo de riesgo: Obesidad?: SI
¿Pertenece al siguiente grupo de riesgo: Embarazo?: SI
¿Pertenece al siguiente grupo de riesgo: Inmunodeficiencia?: SI
¿Pertenece al siguiente grupo de riesgo: Enfermedad renal?: SI
¿Pertenece al siguiente grupo de riesgo: Enfermedad pulmonar?: SI
¿Pertenece al siguiente grupo de riesgo: Cáncer?: SI
¿Pertenece al siguiente grupo de riesgo: Receptor de órganos?: SI
Para culminar, responda las siguientes preguntas:
¿Los últimos 7 días ha tenido contacto con alguien con Covid?: SI
¿Se encuentra vacunado?: SI
¿Cuántas dosis?: 3
¿Ha presentado Covid?: NO
¿Qué tipo de prueba desea?: PCR

Escriba 'SI' si desea ingresar a un paciente a la base datos, caso contrario, escriba 'NO': SI
Ingrese Nombres: Alessandra
Ingrese Apellidos: Oquellis Gonzales
Ingrese Edad: 19
Ingrese Sexo: F
Ingrese Tipo de documento de identidad: DNI
Ingrese Número del documento: 72229224
Ingrese Dirección: La laguna del Chipe H25
Ingrese Número celular: 932558870
Ingrese Correo: oquellis.alessandra@gmail.com
Ingrese Ocupación: Universitario
A continuación deberá escribir 'SI' si ha presentado el síntoma, caso contrario, 'NO'
¿Tiene o ha tenido Tos?: SI
¿Tiene o ha tenido Dolor de garganta?: SI
¿Tiene o ha tenido Congestión nasal?: SI
¿Tiene o ha tenido Dificultad respiratoria?: SI
¿Tiene o ha tenido Fiebre?: SI
¿Tiene o ha tenido Diarrea?: SI
¿Tiene o ha tenido Náuseas/vómitos?: SI
¿Tiene o ha tenido Dolor muscular?: SI
¿Tiene o ha tenido Dolor de pecho?: SI

A continuación deberá escribir 'SI' si pertenece al grupo de riesgo, caso contrario, 'NO'
¿Pertenece al siguiente grupo de riesgo: Mayor de 65?: SI
¿Pertenece al siguiente grupo de riesgo: Enfermedad cardiovascular?: SI
¿Pertenece al siguiente grupo de riesgo: Diabetes?: SI
¿Pertenece al siguiente grupo de riesgo: Enfermedad cerebrovascular?: SI
¿Pertenece al siguiente grupo de riesgo: Síndrome de Down?: SI
¿Pertenece al siguiente grupo de riesgo: Obesidad?: NO
¿Pertenece al siguiente grupo de riesgo: Embarazo?: NO
¿Pertenece al siguiente grupo de riesgo: Inmunodeficiencia?: SI
¿Pertenece al siguiente grupo de riesgo: Enfermedad renal?: NO
¿Pertenece al siguiente grupo de riesgo: Enfermedad pulmonar?: SI
¿Pertenece al siguiente grupo de riesgo: Cáncer?: SI
¿Pertenece al siguiente grupo de riesgo: Receptor de órganos?: SI
Para culminar, responda las siguientes preguntas:
¿Los últimos 7 días ha tenido contacto con alguien con Covid?: SI
¿Se encuentra vacunado?: SI
¿Cuántas dosis?: 3
¿Ha presentado Covid?: NO
¿Qué tipo de prueba desea?: PCR

Escriba 'SI' si desea ingresar a un paciente a la base datos, caso contrario, escriba 'NO': SI

Ingrese Nombres: Henry

Ingrese Apellidos: Fernández Aquino

Ingrese Edad: 19

Ingrese Sexo: M

Ingrese Tipo de documento de identidad: DNI

Ingrese Número del documento: 71646189

Ingrese Dirección: Av.Bolognesi 565

Ingrese Número celular: 937272240

Ingrese Correo: henryfernandezaquino@gmail.com

Ingrese Ocupación: Universitario

A continuación deberá escribir 'SI' si ha presentado el síntoma, caso contrario, 'NO'

¿Tiene o ha tenido Tos?: SI

¿Tiene o ha tenido Dolor de garganta?: SI

¿Tiene o ha tenido Congestión nasal?: SI

¿Tiene o ha tenido Dificultad respiratoria?: SI

¿Tiene o ha tenido Fiebre?: SI

¿Tiene o ha tenido Diarrea?: SI

¿Tiene o ha tenido Náuseas/vómitos?: SI

¿Tiene o ha tenido Dolor muscular?: SI

¿Tiene o ha tenido Dolor de pecho?: SI

A continuación deberá escribir 'SI' si pertenece al grupo de riesgo, caso contrario, 'NO'

¿Pertenece al siguiente grupo de riesgo: Mayor de 65?: SI

¿Pertenece al siguiente grupo de riesgo: Enfermedad cardiovascular?: SI

¿Pertenece al siguiente grupo de riesgo: Diabetes?: SI

¿Pertenece al siguiente grupo de riesgo: Enfermedad cerebrovascular?: SI

¿Pertenece al siguiente grupo de riesgo: Síndrome de Down?: SI

¿Pertenece al siguiente grupo de riesgo: Obesidad?: SI

¿Pertenece al siguiente grupo de riesgo: Embarazo?: NO

¿Pertenece al siguiente grupo de riesgo: Inmunodeficiencia?: SI

¿Pertenece al siguiente grupo de riesgo: Enfermedad renal?: SI

¿Pertenece al siguiente grupo de riesgo: Enfermedad pulmonar?: SI

¿Pertenece al siguiente grupo de riesgo: Cáncer?: SI

¿Pertenece al siguiente grupo de riesgo: Receptor de órganos?: SI

Para culminar, responda las siguientes preguntas:

¿Los últimos 7 días ha tenido contacto con alguien con Covid?: SI

¿Se encuentra vacunado?: SI

¿Cuántas dosis?: 2

¿Ha presentado Covid?: NO

¿Qué tipo de prueba desea?: PCR

Escriba 'SI' si desea ingresar a un paciente a la base datos, caso contrario, escriba 'NO': SI

Ingrese Nombres: Vincenzo

Ingrese Apellidos: Luna Otero

Ingrese Edad: 19

Ingrese Sexo: M

Ingrese Tipo de documento de identidad: DNI

Ingrese Número del documento: 75960558

Ingrese Dirección: Av.San Martin 345

Ingrese Número celular: 954107586

Ingrese Correo: lunaoteroov@gmail.com

Ingrese Ocupación: Universitario

A continuación deberá escribir 'SI' si ha presentado el síntoma, caso contrario, 'NO'

¿Tiene o ha tenido Tos?: SI

¿Tiene o ha tenido Dolor de garganta?: SI

¿Tiene o ha tenido Congestión nasal?: SI

¿Tiene o ha tenido Dificultad respiratoria?: SI

¿Tiene o ha tenido Fiebre?: SI

¿Tiene o ha tenido Diarrea?: SI

¿Tiene o ha tenido Náuseas/vómitos?: SI

¿Tiene o ha tenido Dolor muscular?: SI

¿Tiene o ha tenido Dolor de pecho?: SI

A continuación deberá escribir 'SI' si pertenece al grupo de riesgo, caso contrario, 'NO'

¿Pertenece al siguiente grupo de riesgo: Mayor de 65?: SI

¿Pertenece al siguiente grupo de riesgo: Enfermedad cardiovascular?: SI

¿Pertenece al siguiente grupo de riesgo: Diabetes?: SI

¿Pertenece al siguiente grupo de riesgo: Enfermedad cerebrovascular?: SI

¿Pertenece al siguiente grupo de riesgo: Síndrome de Down?: SI

¿Pertenece al siguiente grupo de riesgo: Obesidad?: SI

¿Pertenece al siguiente grupo de riesgo: Embarazo?: SI

¿Pertenece al siguiente grupo de riesgo: Inmunodeficiencia?: SI

¿Pertenece al siguiente grupo de riesgo: Enfermedad renal?: SI

¿Pertenece al siguiente grupo de riesgo: Enfermedad pulmonar?: SI

¿Pertenece al siguiente grupo de riesgo: Cáncer?: SI

¿Pertenece al siguiente grupo de riesgo: Receptor de órganos?: SI

Para culminar, responda las siguientes preguntas:

¿Los últimos 7 días ha tenido contacto con alguien con Covid?: NO

¿Se encuentra vacunado?: SI

¿Cuántas dosis?: 2

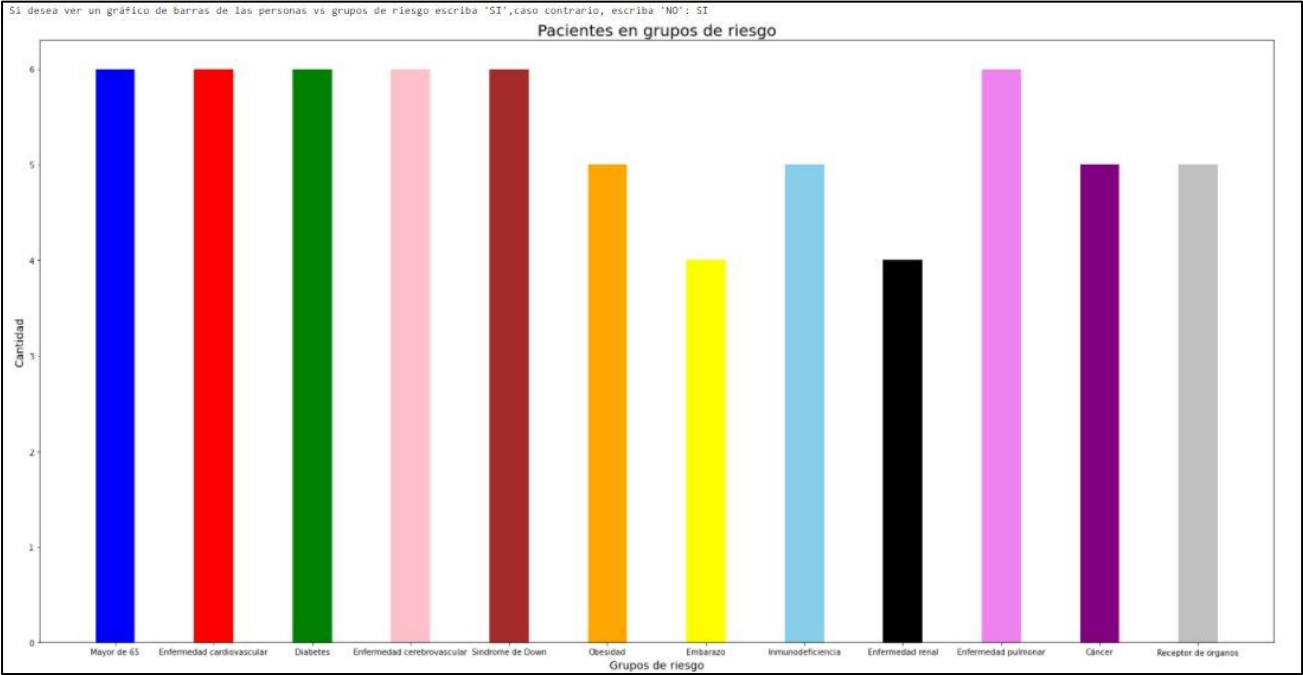
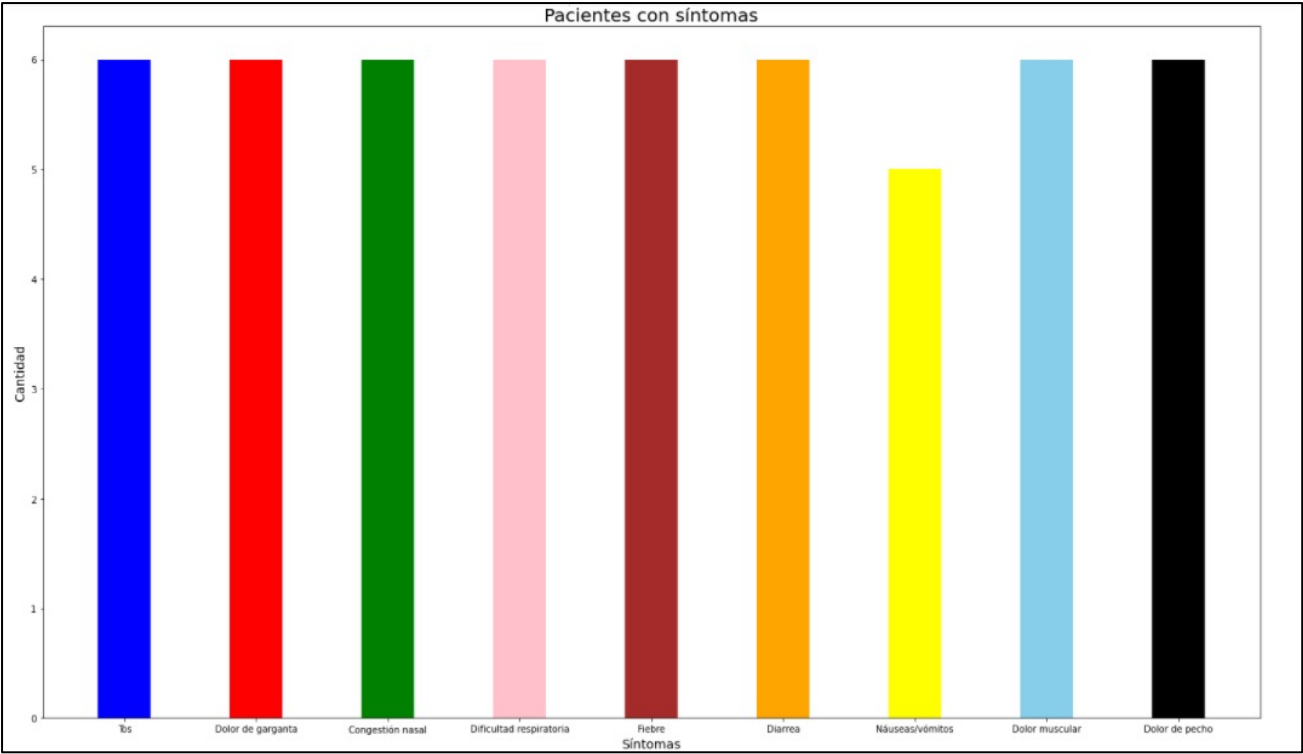
¿Ha presentado Covid?: SI

¿Qué tipo de prueba desea?: PCR

Escriba 'SI' si desea ingresar a un paciente a la base datos, caso contrario, escriba 'NO': NO

Necesita agregar los datos personales para continuar con el registro

SI desea ver un gráfico de barras de las personas vs síntomas escriba 'SI', caso contrario, escriba 'NO': SI



Si desea Filtrar datos según cierto criterio escriba SI, caso contrario, escriba NO: SI																						
Ingrese el valor del encabezado a filtrar: Nombres																						
Ingrese el valor del filtro: Bruno																						
Nombres		Apellidos	Edad	Sexo	Tipo de documento de identidad	Número del documento	Dirección		Número celular	Correo	Ocupación	Tos	Dolor de garganta	Congestión nasal	Dificultad respiratoria	Fiebre	Diarrea	Náuseas/vómitos	Dolor muscular	Dolor de pecho		
0	Bruno	Calopña Arbulú	19	M	DNI	74159607	Passaje los Sauces 120		986220580	brunocalopi@gmail.com	Universitario	SI	SI	SI	SI	SI	SI	SI	NO	SI		
Nombres		Apellidos	Edad	Sexo	Tipo de documento de identidad	Número del documento	Dirección		Número celular	Correo	Ocupación	Enfermedad cardiovascular	Diabetes	Enfermedad cerebrovascular	Síndrome de Down	Obesidad	Embarazo	Inmunodeficiencia	Enfermedad renal	Enfermedad pulmonar	Cáncer	Receptor de órganos
0	Bruno	Calopña Arbulú	19	M	DNI	74159607	Passaje los Sauces 120		986220580	brunocalopi@gmail.com	Universitario	SI	SI	SI	SI	SI	SI	SI	NO	NO	SI	NO
Si desea Filtrar datos según cierto criterio escriba SI, caso contrario, escriba NO: Apellidos																						
No se ha filtrado																						

4. Conclusiones

- Gracias a Python hemos podido introducirnos a la programación, siendo nuestro aliado de ahora en adelante para nuestros trabajos de investigación tratando una determinada base de datos.
- El software nos ha permitido poder expresar en distintas formas nuestras ideas con la finalidad de analizar y almacenar una determinada base de datos, permitiendo reflejar de una forma sencilla y selecta en su forma algorítmica.
- El programa nos ayudó a poder aprender nuevos comandos y librerías que nos permitan profundizar su estudio y las distintas mejoras que le podemos dar nosotros como creadores de los programas.
- Nos permitió introducirnos en la automatización de procesos, curso que llevaremos los ingenieros mecánicos eléctricos (Sistemas Automáticos de Control), en donde tendremos un controlador que partirá de datos, teniendo como responsabilidad qué acción de control será necesaria y realizar la regulación para mantener el sistema en determinados valores.
- El presente trabajo nos ha permitido poder realizar un programa para minimizar procesos y de esa manera innovar en un centro de salud, en donde se aplican pruebas ante la Covid – 19. Gracias a ello, hemos logrado almacenar la base de datos necesarios para evaluar estratégicamente la opción más eficaz y eficiente de los pacientes.
- En conclusión, Python nos ha permitido analizar de manera detallada una base de datos y a la vez mejorar los determinados códigos para optimizar dichos procesos ya que esta herramienta nos permite realizar modificaciones de una manera automatizada.