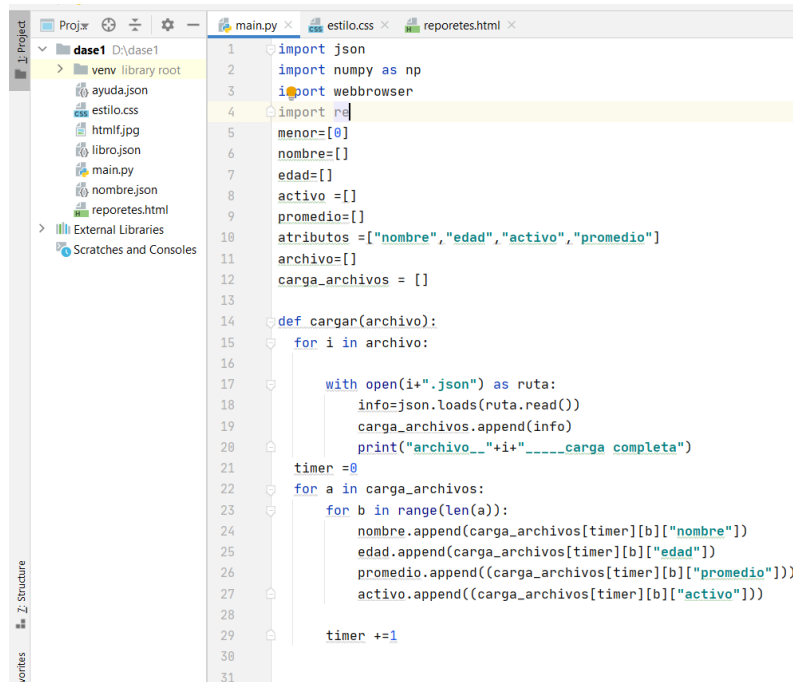


Manual técnico

Para realizar este proyecto se utilizaron tres archivos json de prueba los cuales fueron nombre, ayuda, libro para empezar a programar se importaron tres librerías la cuales son json, numpy as np y webbrowser así mismo se declararon cuatro variables.



```
1 import json
2 import numpy as np
3 import webbrowser
4 import re
5 menor=[0]
6 nombre=[]
7 edad=[]
8 activo =[]
9 promedio=[]
10 atributos=["nombre", "edad", "activo", "promedio"]
11 archivo=[]
12 carga_archivos = []
13
14 def cargar(archivo):
15     for i in archivo:
16
17         with open(i+".json") as ruta:
18             info=json.loads(ruta.read())
19             carga_archivos.append(info)
20             print("archivo_"+i+"-----carga completa")
21         timer =0
22         for a in carga_archivos:
23             for b in range(len(a)):
24                 nombre.append(carga_archivos[timer][b]["nombre"])
25                 edad.append(carga_archivos[timer][b]["edad"])
26                 promedio.append((carga_archivos[timer][b]["promedio"]))
27                 activo.append((carga_archivos[timer][b]["activo"]))
28
29             timer +=1
30
31
```

Después de todo esto empezamos a programar nuestra primera función que seria la de carga de archivos donde nuestro programa va leer la palabra “carga” seguido de los archivos después de todo eso se manda a nuestra segunda función que es donde los archivos se cargan a memoria la segunda función def carga(archivo): y ese seria para nuestro comando cargar

```

def main():

    print("ingrese ruta")
    ingresar=input().split(",")
    archivo = ingresar
    comand = ingresar[0][:6]
    archivo[0] = archivo[0].replace(" ",",")
    archivo[0] = archivo[0][6:]
    if(comand.lower()=="cargar"):
        for a in range(len(archivo)):
            archivo[a]= archivo[a].replace(" ",",")
    cargar(archivo)
    menu()

```

Después se hizo una función llamada menú en el cual se encontrar un menú en donde estarán todos los comandos en el que el usuario puede acceder

```

def menu():
    print("menu")
    opciones = input()
    while opciones != "salir":
        if opciones == "seleccionar":
            siguiente()
        if opciones == "maximo":
            maximos()
        if opciones == "minimo":
            minimo()
        if opciones == "suma":
            sedades()
        if opciones == "cuenta":
            scuentas()
        if opciones == "reporte":
            reporte()

```

Para el comando seleccionar se hizo una función llamada seleccionar en cual consiste en que va recorrer cada una de las listas de nuestros archivos hasta encontrar el nombre, edad, promedio o el activo seleccionado después que encuentre una de estas opciones el usuario debe decidir que atributos desea imprimir

```
def siguiente():
    opcion1 = input()
    opcion2 = input()
    opcion3 = input()
    opcion4 = input()
    opcion5 = input()

    #-----NOMBRE[0][0]-----
    if(opcion1==carga_archivos[0][0]["nombre"]):
        if(opcion2 == "nombre"):
            print(carga_archivos[0][0]["nombre"])
        if(opcion3 == "edad"):
            print(carga_archivos[0][0]["edad"])
        if(opcion4=="promedio"):
            print(carga_archivos[0][0]["promedio"])
        if(opcion5=="activo"):
            print(carga_archivos[0][0]["activo"])

    #-----EDAD[0][0]-----
    a =str(carga_archivos[0][0]["edad"])
    if (opcion1 ==a):
        if (opcion2 == "nombre"):
            print(carga_archivos[0][0]["nombre"])
        if (opcion3 == "edad"):
            print(carga_archivos[0][0]["edad"])
        if (opcion4 == "promedio"):
            print(carga_archivos[0][0]["promedio"])
        if (opcion5 == "activo"):
            print(carga_archivos[0][0]["activo"])

    #-----PROMEDIO[0][0]-----
```

Después en la función máximo y mínimo se utilizara numpy para poder recorrer las listas e imprimir el máximo promedio o el máximo edad así mismo con el mínimo edad y promedio

```
def maximos():
    print("entro")
    maxE = input()
    if(maxE=="maximo edad"):
        maximiso = np.array(edad)
        print("Edad maxima es >>>>>" +str(np.amax(maximiso)))
    maxD = input()
    if (maxD == "maximo promedio"):
        maximiso = np.array(promedio)
        print("promedio maxima es >>>>>" + str(np.amax(maximiso)))
    menu()

def minimo():
    print("entro")
    maxE = input()
    if (maxE == "minimo edad"):
        maximiso = np.array(edad)
        print("Edad minima es >>>>>" + str(np.amin(maximiso)))
    maxD = input()
    if (maxD == "minimo promedio"):
        maximiso = np.array(promedio)
        print("promedio minimo es >>>>>" + str(np.amin(maximiso)))
    menu()
```

Después usaremos dos funciones para imprimir la suma de edades y suma promedios así mismo para ver el numero total de registros que existen en nuestros archivos json para eso también utilizamos numpy

```
def sedades():
    print("entro")
    ayuda = input()
    if ayuda == "suma edad":
        edades(edad)
    spromedio()

def spromedio():
    ayuda = input()
    if ayuda == "suma promedio":
        promedios(promedio)
    menu()

def scuentas():
    ayuda = input()
    if ayuda == "cuenta":
        print("numero de registro>>" + str(len(nombre)))
    menu()
```

Y por ultimo la función reportes en el cual emplearemos html para poder crear una tabla con ciertos registros seleccionados así mismo utilizaremos css para ser mas atractivo la pagina web para poder leer el documento html se utilizo la Liberia webbrowser

```
def reporte():
    # ' ' \

    numero1 = input()
    numero = int(numero1)

    css = '<!DOCTYPE html> \n <html lang = "en"> \n <head> \n <meta charset = "utf-8">\n<title>Reporte</title>\n' + '<link rel="stylesheet" href="estilo.css">'
    css = css + '</head>\n<body>\n<center>\n<table border = "1">\n<tr> '
    for element in range(len(atributos)):
        nmbs = '<th bgcolor="blue">' + atributos[element] + '</th>'
        css = css + nmbs

    for element2 in range(numero):
        ayuda = '<tr> \n <td>' + str(nombre[element2]) + '</td> <td>' + str(edad[element2]) + '</td> <td>' + str(promedio[element2]) + '</td> <td>' + str(activo[element2])
        ayuda = ayuda + '\n</tr>\n '
        css = css + ayuda
    css = css + '</table> \n </center> \n </body>\n </html> '
    crate = open("reporetes.html", "w")
    crate.write(css)
    crate.close()
    webbrowser.open_new_tab("reporetes.html")
```

