

8-2-2020

MT

MANUAL TECNICO SIMPLE SQL



OBJETIVO GENERAL

Entender y aplicar los conceptos básicos de los lenguajes de programación, así como con los conceptos de los paradigmas a los que estos pertenecen

OBJECTIVOS ESPECIFICOS

1. Implementar una solución de software utilizando Python como lenguaje.
2. Implementar una solución en consola que utilice el paradigma declarativo.
3. Crear una aplicación con la capacidad de cargar información en formato json y entregar reportes atractivos utilizando HTML.

TECNOLOGIAS

PYTHON

HTML

CSS3

BOOSTRAP4

JAVASCRIPT

VERSIONES

PYTHON 3 (version 3.8.5)

HTML (version 5)

CSS (version 3)

BOOSTRAP (version 4)

JAVASCRIPT (version 6)

EDITOR UTILIZADO

VISUAL STUDIO CODE

```

from argparse import ArgumentParser
from pathlib import Path
import json
from string import Template
import webbrowser as wb
archivoHTML = open("index.html")
template = Template(archivoHTML.read())
print ("-----")
print ("| SIMPLE SQL |")
print ("-----")
print ("INGRESE LOS ARCHIVO O ARCHIVO RECUERDE PONER LA EXTENSION DEL ARCHIVO (.json)")

```

- 1) Se importaron los módulos necesarios para realizar la aplicación donde se importó json para poder manipular los archivos json también se utilizó Template para poder desplegar la pagina web

```

cargarDatos = input()

def cargar(datos):
    Cargando...
    if datos.find("CARGAR") == 0:
        reemplazable = datos.replace("CARGAR", "")
        reemplazable2 = reemplazable.replace(", ", "")
        print (reemplazable2.split())
        for nombreArchivo in reemplazable2.split():
            with open(nombreArchivo) as archivoJson:
                jsonArchivo = json.load(archivoJson)
                print("SE HA CARGADO EL ARCHIVO")
    elif cargarDatos.find("SALIR") == 0:
        exit()

cargar(cargarDatos)

```

- 2) Se creó un input para poder obtener los datos ingresados este input colecciona los nombres de los archivos, se creó una función cargar con un parámetro llamado datos este parámetro obtiene el nombre de los archivos.

La funcionalidad de la función es buscar la palabra cargar si encuentra la palabra cargar reemplaza esa palabra por un string vacío, luego se recorre los archivos con un for para obtener la información de los archivos json.

```

30
31 def procesos(datos):
32     reemplazable = datos.replace("CARGAR","")
33     reemplazable2 = reemplazable.replace(",","")
34     while True:
35         proceso = input()
36
37         if proceso.find("SELECCIONAR") == 0:
38             if proceso.find("SELECCIONAR *")== 0:
39                 for nombreArchivo in reemplazable2.split():
40                     with open(nombreArchivo) as archivoJson:
41                         jsonArchivo = json.load(archivoJson)
42                         for linea in jsonArchivo:
43                             print ("nombre: "+str(linea.get("nombre")))
44                             print ("edad "+str(linea.get("edad")))
45                             print ("activo: "+str(linea.get("activo")))
46                             print ("promedio: "+ str(linea.get("promedio")))
47             elif proceso.find("SELECCIONAR nombre, edad, promedio, activo DONDE nombre = ")==0:
48
49                 nuevo = proceso.replace("SELECCIONAR nombre, edad, promedio, activo DONDE nombre = ","")
50                 sincomillas = nuevo.strip('')
51
52                 for nombreAr nombreArchivo: str 2.split():
53                     with open(nombreArchivo) as archivoJson:
54                         jsonArchivo = json.load(archivoJson)
55                         for linea in jsonArchivo:
56                             if linea.get("nombre") == sincomillas:
57                                 print ("nombre: "+str(linea.get("nombre")))
58                                 print ("edad "+str(linea.get("edad")))
59                                 print ("activo: "+str(linea.get("activo")))
60                                 print ("promedio: "+ str(linea.get("promedio")))
61             elif proceso.find("SELECCIONAR nombre, edad DONDE promedio")==0:
62                 nuevo2 = proceso.replace("SELECCIONAR nombre, edad DONDE promedio = ","")
63                 for nombreArchivo in reemplazable2.split():
64                     with open(nombreArchivo) as archivoJson:
65                         jsonArchivo = json.load(archivoJson)
66                         for linea in jsonArchivo:
67                             if str(linea.get("promedio")) == nuevo2 :
68                                 print ("nombre: "+str(linea.get("nombre")))
69                                 print ("edad: "+str(linea.get("edad")))
70             elif proceso.find("MAXIMO edad") == 0:
71                 lista =[]
72                 n =0
73                 for nombreArchivo in reemplazable2.split():

```

- 3) Se creó una función llamada procesos con un parámetro llamado datos donde ese parámetro recibe los nombres de los archivos .

Se creó una variable con el nombre reemplazable esta variable reemplaza la palabra CARGAR en un espacio en blanco esto se hizo con la función replace.

Se creó una variable con el nombre reemplazable2 donde esta variable obtiene los datos de la variable reemplazable donde se reemplazara la comilla que trae el string.

Se creó un while con un valor true al iniciar el ciclo while se crea un input donde se puede ingresar los comandos.

Se utilizó la función find para buscar la palabra “SELECCIONAR” si la encuentra se procede a realizar el proceso que esta dentro de la condición if, dentro del if se crea un for donde recorrerá los archivos json y poder

obtener los campos que se necesitan luego de obtener los datos se muestran en consola.

Se creó un elif en la línea 47 donde esto hace una validación con el método find si cumple la validación se ejecuta un método replace donde este método reemplaza todos los valores que no nos reviran y solo deja el nombre del registro que queremos buscar luego se utilizó el método strip para eliminar las comillas del nombre para poder comparar con los otros datos y regresar los datos correctos

Se crea un for donde este for recorre los datos de los archivos y los imprime en pantalla

```
75     jsonArchivo = json.load(archivoJson)
76
77     for linea in jsonArchivo:
78         lista.append(linea.get("edad"))
79     print(max(lista))
80 elif proceso.find("MAXIMO promedio") == 0:
81     listaPromedio = []
82     for nombreArchivo in reemplazable2.split():
83         with open(nombreArchivo) as archivoJson:
84             jsonArchivo = json.load(archivoJson)
85
86             for linea in jsonArchivo:
87                 listaPromedio.append(linea.get("promedio"))
88     print(max(listaPromedio))
89 elif proceso.find("MINIMO edad") == 0:
90     listaEdad = []
91     for nombreArchivo in reemplazable2.split():
92         with open(nombreArchivo) as archivoJson:
93             jsonArchivo = json.load(archivoJson)
94
95             for linea in jsonArchivo:
96                 listaEdad.append(linea.get("edad"))
97     print(min(listaEdad))
98 elif proceso.find("MINIMO promedio") == 0:
99     listaPromedioMinimo = []
100    for nombreArchivo in reemplazable2.split():
101        with open(nombreArchivo) as archivoJson:
102            jsonArchivo = json.load(archivoJson)
103
104            for linea in jsonArchivo:
105                listaPromedioMinimo.append(linea.get("promedio"))
106    print(min(listaPromedioMinimo))
```

- 4) Se cargaron los archivos json para guardarlos en memoria se creó un for donde se obtiene los campos para mostrarlos en consola donde se utilizó el método max para poder obtener la edad mayor de todos los registros de los archivos

Se creó otro for para obtener los registros y poder encontrar el mayor promedio por medio del método max.

Se creo un elif donde por el cual se valida con un método find si la oracion MINIMO edad es encontrada se ejecuta la sentencia en verdadera , se crea una lista de arreglos donde se meterán los números para obtener luego con el método min el número menor del arreglo

Con el MINIMO promedio se realizó lo mismo que el MINIMO edad.

```
117     print(incremento)
118     elif proceso.find("SUMA promedio") == 0:
119         incrementos=0
120         listaSumaPromedio =[]
121         for nombreArchivo in remplazable2.split():
122             with open(nombreArchivo) as archivoJson:
123                 jsonArchivo = json.load(archivoJson)
124                 for linea in jsonArchivo:
125                     listaSumaPromedio.append(linea.get("promedio"))
126         for promedio in listaSumaPromedio:
127             incrementos = incrementos + promedio
128         print(incrementos)
129     elif proceso.find("CUENTA") == 0:
130         incrementoCuenta=0
131         listaSumaPromedio =[]
132         for nombreArchivo in remplazable2.split():
133             with open(nombreArchivo) as archivoJson:
134                 jsonArchivo = json.load(archivoJson)
135                 for linea in jsonArchivo:
136                     incrementoCuenta = 1 + incrementoCuenta
137         print(incrementoCuenta)
138     elif proceso.find("REPORTE")== 0:
139         incrementoNumero=0
140         nombres = ""
141         edad = ""
142         activo = ""
143         promedio = ""
144         numero = proceso.replace("REPORTE ", "")
145         for nombreArchivo in remplazable2.split():
146             with open(nombreArchivo) as archivoJson:
147                 jsonArchivo = json.load(archivoJson)
148                 for linea in jsonArchivo:
```

- 5) se creo un elif donde se validara verdadero si el metodo find encuentra la palabra “suma promedio” , si encuentra la palabra se creara un arreglo vacio, luego se creo un for para recorrer los campos, se creo una variable contador donde esta variable ira sumando todas las edades hasta obtener la toda la suma de las edades para mostrar los datos se utilizo un print.

Se realizó lo mismo con la suma promedio.

Se creó un elif para validar si el metodo find encuentra la palabra “CUENTA” se crea un arreglo vacio luego del arreglo vacio se crea un for donde recorre los datos de los documentos , luego se creo una variable

incrementacuenta donde esta incrementa en 1 la cantidad de veces que recorre el for luego esa variable incrementacuenta se imprime en consola con el metodo print

Se creó un elif donde se utiliza el metodo find para verificar si la palabra existe y si existe se ejecuta la sentencia en verdadero, se crean variables locales nombre, edad, activo y promedio

Se utiliza replace para quitar de la oracion la palabra "REPORTE " y se reemplaza por una cadena vacia, se creó un un for para recorrer los campos.

Se utilizo el metodo del modulo String Template llamado load para poder cargar los archivos en un html ya creado.

```
with open(nombreArchivo) as archivoJson:
    jsonArchivo = json.load(archivoJson)
    for linea in jsonArchivo:
        if int(numero)-1>=incrementoNumero:
            nombres = nombres + str(linea.get("nombre"))+ " "
            edad = edad + str(linea.get("edad")) + " "
            activo = activo + str(linea.get("activo")) + " "
            promedio = promedio + str(linea.get("promedio")) + " "
            incrementoNumero = 1+ incrementoNumero

for i in range(int(numero)):
    if int(numero) == 1:
        file2 = open("json.html","w")
        resultado = temple.substitute({'nombre':nombres, 'edad': edad, 'activa':activo, 'promedio': promedio})
        file2.writelines(resultado)
        file2 = open("json.html","w")
        resultado = temple.substitute({'nombre':nombres, 'edad': edad, 'activa':activo, 'promedio': promedio})
        file2.writelines(resultado)
    wb.open_new("json.html")
elif proceso.find("SALIR")==0:
    exit()
```

- 6) Se creó una condición donde si numero -1 es mayor o igual a incremento numero esto imprimirá el número de datos que quiere el cliente en el reporte luego se utilizó otro for para poder recorrer los datos y mostrarlos en el documento HTML

Se creó un comando extra para poder terminar la aplicación utilizando el método exit que es nativo de Python.