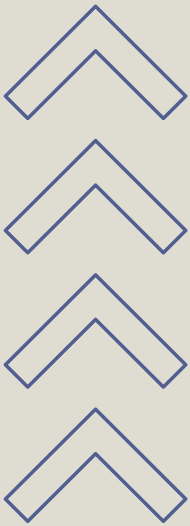


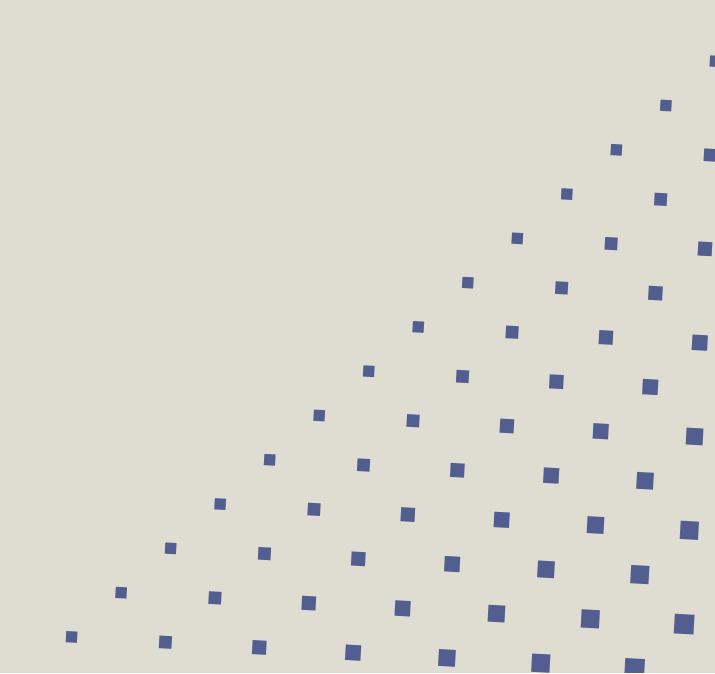


**LENGUAJES FORMALE DE
PROGRAMACIÓN**

MANUAL TÉCNICO




Informe redactado por:
Jose Fuentes
202001228





ÍNDICE

• Introducción	1
• Información sobre el programa	1
• Importación del programa	2
• Variables globales	2
• Clases	2
• Carga Archivo	3
• Gestion de Cursos	4
• Agregar Cursos	4
• Conteo de Créditos	5





➤➤➤➤➤ INTRODUCCIÓN

Este es un programa realizado para evaluar los conocimientos obtenidos por los estudiantes en cursos anteriores. Manejando arreglos, variables, clases, objetos e interfaz grafica. Con el objetivo de explicar por el código para posibles modificaciones del mismo, explicando cada uno de los métodos y funciones utilizadas. Mejorando la experiencia del usuario y proporcionando la ayuda necesaria para el mismo.

➤➤➤➤➤ INFORME SOBRE EL PROGRAMA

Este programa fue realizado en el lenguaje de Python versión 3.10 con la cual se desarrollo un software que ayudara al estudiante a saber como va en la carrera mediante los cursos aprobados, cursando y pendientes que tiene el estudiante. Por medio de agregación de cursos y con una carga masiva para facilitar el ingreso de cada uno de los cursos.

Este es un sistema que 64 bits, se puede utilizar en los siguientes sistemas operativos:

- Windows 7
- Windows 8
- Windows 8.1
- Windows 10
- Windows 11

Requisitos para el uso del programa:

- 1 GB de ram
- Espacio en disco duro de 256 mb
- Procesador: Minimo Pentium hasta gama alta de procesadores



IMPORTACIONES DEL PROGRAMA

Al iniciar la programación de este programa, se realizaron 4 importaciones para la mejor experiencia a dar al usuario.

1. tkinter: extensión que sirve para elaboración de interfaz
2. Cursos: archivo en el cual se declaró el objeto para la manipulación de los datos ingresados por el usuario.
3. messagebox: para enviar mensajes de advertencia, información o error al usuario a la hora de estar manipulando el programa.

Se adjunta una foto y las importaciones se encontraron en las primeras líneas del código.

```
from tkinter import *  
from tkinter import ttk  
from Cursos import Cursos  
from tkinter import messagebox
```



VARIABLE GLOBAL

En este programa solo se declaró una sola variable global, la cual servirá para almacenar los distintos cursos ingresados por los usuarios. La cual llevará el nombre de cursos y se declara como una lista.

```
global cursos  
cursos = []
```



CLASES

En el programa hay diferentes clases que hacen referencia a cada una de las ventas que se utilizarán en el desarrollo de todo el programa, cada clase tiene diferentes funciones y métodos que ayudarán a la resolución de cada uno de los incisos de cada ventana.

- Menu()
- CargaArchivo()
- GestionarCurso

Las siguientes clases hacen referencia al sub menú de Gestión de Curso en el cual se harán las funciones de cada una de las clases.

- ListaCursos
- AgregarCurso
- EditarCurso
- Eliminar

Esta clase es la ultima donde el usuario podrá interactuar con los datos ingresados anteriormente.

- ConteoCreditos

»»»» CARGA ARCHIVO

En esta clase se encuentra el código para poder ingresar un archivo con extencion .Ifp el cual servirá para ingresar con carga masiva los cursos de un estudiante.

Por cuestiones de estética no se colocara todo el código, pero estará la parte mas importante que lee el archivo con la ruta que se ingreso y lo almacena en una lista para poder guardar todo el contenido del archivo.

```
try:
    objeto = open(ruta,'r+',encoding='utf-8')
    lineas = objeto.readlines()
    objeto.close()

    for linea in lineas:
        data = linea.split(',')
        print(data)
        curso = Cursos(data[0],data[1],data[2],data[3],data[4],data[5],
data[6].rsplit('\n'))
        cursos.append(curso)

    return cursos

except:
    messagebox.showerror("Error", "El archivo ingresado no es valido \n
Intente de nuevo :)")
    self.ruta.delete("1.0","end")
```



GESTIÓN DE CURSOS

En esta clase se encontrara un menú que redireccionara a otras clases con el objetivo de ver la lista de cursos ingresados, eliminacion, edición y agregar un curso manualmente.



AGREGAR CURSO

En esta clase se encontrara un menú que redireccionara a otras clases con el objetivo de ver la lista de cursos ingresados, eliminación, edición y agregar un curso manualmente. Esta es parte del código por cuestión de estética no se mostrara todo del código para agregar un curso mediante unas función que se llama agregarCurso.

```
if (analizador == 0):
    cursos = self.agreCurso(codigoA,nombreA,requisitoA,semestreA,opcionalA,
    creditosA,estadoA)
    for curso in cursos:
        print(curso.getCodigo()," ",curso.getNombre())

    self.codigo.delete("1.0","end")
    self.nombre.delete("1.0","end")
    self.requisito.delete("1.0","end")
    self.semestre.delete("1.0","end")
    self.opcional.delete("1.0","end")
    self.creditos.delete("1.0","end")
    self.estado.delete("1.0","end")
    messagebox.showinfo("Información", "Curso Agregado con éxito.")
```

CONTEO DE CRÉDITOS

En esta parte del código se mostrara lo de créditos aprobados automáticamente, créditos cursando y pendientes. En la parte de abajo de mostrara el un comboBox en el cual seleccionara el numero de semestre que quiere evaluar y le mostrara la cantidad de créditos que lleva hasta ese momento.

```
def conteoObligatorios(self):
    try:
        seleccionado = self.combo.get()
        contarCreditos = 0
        limite = int(seleccionado)
        if seleccionado != "":
            for curso in cursos:
                if curso.getOpcional() == "1":
                    if (int(curso.getSemestre()) <= limite):
                        contarCreditos = int(curso.getCreditos()) + contarCreditos
            else:
                messagebox.showerror("Error", "Seleccione un semestre a evaluar.\nO valor ingresado no obtenido en la base de datos.")

        self.obligatirios['text'] = contarCreditos
    except:
        messagebox.showerror("Error", "Ningun valor seleccionado")
```

El conteo de créditos por semestre se hará de la misma manera de conteo de créditos obligatorios, con la selección del semestre deseado y se mostrara tanto en la parte de arriba todos los créditos que están aprobados, cursando y pendientes.

```
try:
    seleccionado = self.combo1.get()
    print("usted selecciono ",seleccionado)
    contadorAprobados = 0
    contadorCursando = 0
    contadorPendiente = 0

    for curso in cursos:
        if curso.getOpcional() == "0":
            contadorAprobados = int(curso.getCreditos()) + contadorAprobados
        elif curso.getOpcional() == "1":
            contadorCursando = contadorCursando + int(curso.getCreditos())
        elif curso.getOpcional() == "-1":
            contadorPendiente = contadorPendiente + int(curso.getCreditos())

    self.aprobadosLabel['text'] = contadorAprobados
    self.cursandoLabel['text'] = contadorCursando
    self.pendientesLabel['text'] = contadorPendiente

except:
    messagebox.showerror("Error", "Ningun valor seleccionado")
```