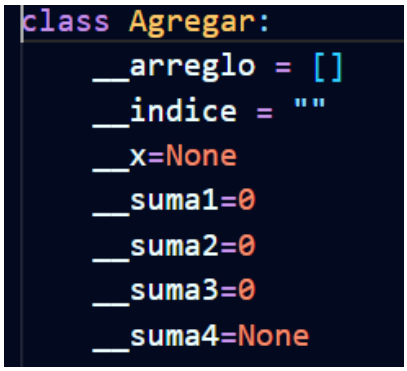
Manual Técnico

Se utilizo el paradigma de programación orientado a objetos ya que cada curso representa un objeto de la vida real.

El programa fue realizado en el lenguaje de programación Python, en su versión 3.10.3, utilizando el editor de código Visual Studio Code

Para la creación del programa se realizaron 3 archivos:

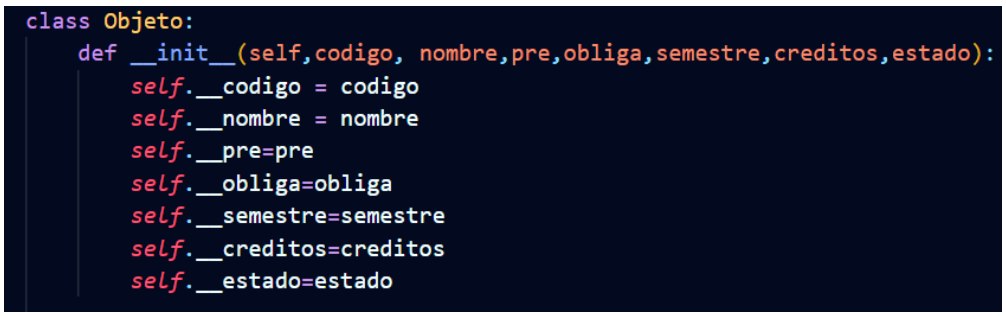
1. Clase Agregar:
   Esta se encarga de toda lo lógica del programa como agregar las instancias a la lista, buscar dentro de la lista, eliminar un curso de la lista asi como realizar las respectivas sumas de créditos solicitados.



2. Clase Objeto:
   Crea las instancias con las distintas características de un curso tales como código, nombre, etc.

3. Archivo main:
   Contiene toda la lógica de la interfaz del programa haciendo uso de la programación funcional para cada una de las ventanas.

Para la realización de la interfaz gráfica se utilizó la librería nativa de Python Tkinter, la cual es fácil y rápida de implementar.



Para llamar a las cajas de texto (Entry) desde una función externa se enviaron como argumentos dichos elementos cuando se llama a la función que ejecutará, por ejemplo:

El método para la suma de créditos deberá insertar los valores en dichas cajas, pero este al recibirlas como parámetros podemos hacer uso de estas.

```
def credHasta1(sem, in3,in4,in5):
    if sem>10 or sem<=0:
        messagebox.showerror(title="Sem
        return
    (c1,c2,c3)=__clase.sumarCred2(sem)
    in3.delete(0,"end")
    in4.delete(0,"end")
    in5.delete(0,"end")
    in3.insert(0, c1)
    in4.insert(0, c2)
    in5.insert(0, c3)
    c1=""
    c2=""
    c3=""
    sem=""
```

Para ocultar ventanas y mostrar otras se implementó la función lambda que permite ejecutar funciones de una sola línea, dicha función se ejecuta dentro del atributo del botón “*command”*.

```
buscar=Button(agregar, text="Buscar",command=lambda:Busqueda(in1.get(),in2))
buscar.place(x=400,y=15)
```