**Reto 4**

**Jorvin A. Zapata Hinestroza**

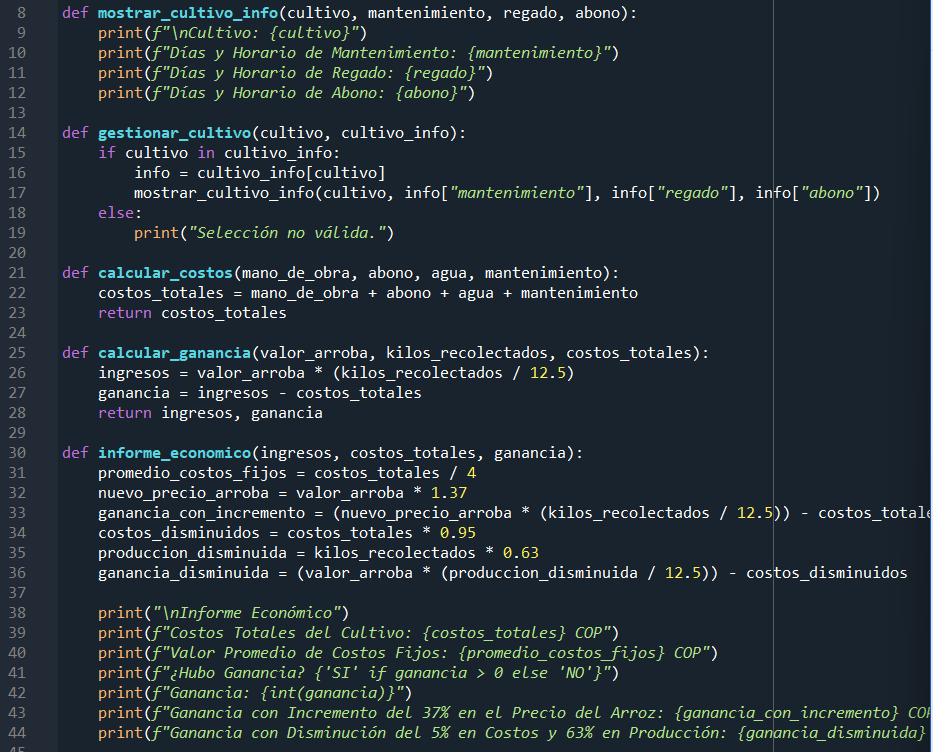
**Punto 1 reto 4**

Realicé la segunda versión del reto 3 pero no utilicé funciones ya que tenía entendido que este se haría solo con el contenido que habíamos visto hasta esa semana (3). Por lo tanto, a continuación, explicaré como pude haber aplicado el paradigma funcional, funciones que pude haber hecho, y los pasos a seguir para crear la solución bajo este paradigma.

**Descripción del código del Reto 3:** el código que hice en el reto 3 basé todas las operaciones y la lógica a través de estructuras de bucles y condicionales. Sin embargo, explicaré por pasos cómo podría utilizar el paradigma funcional para que mi código sea mas orgánico, modular y legible.

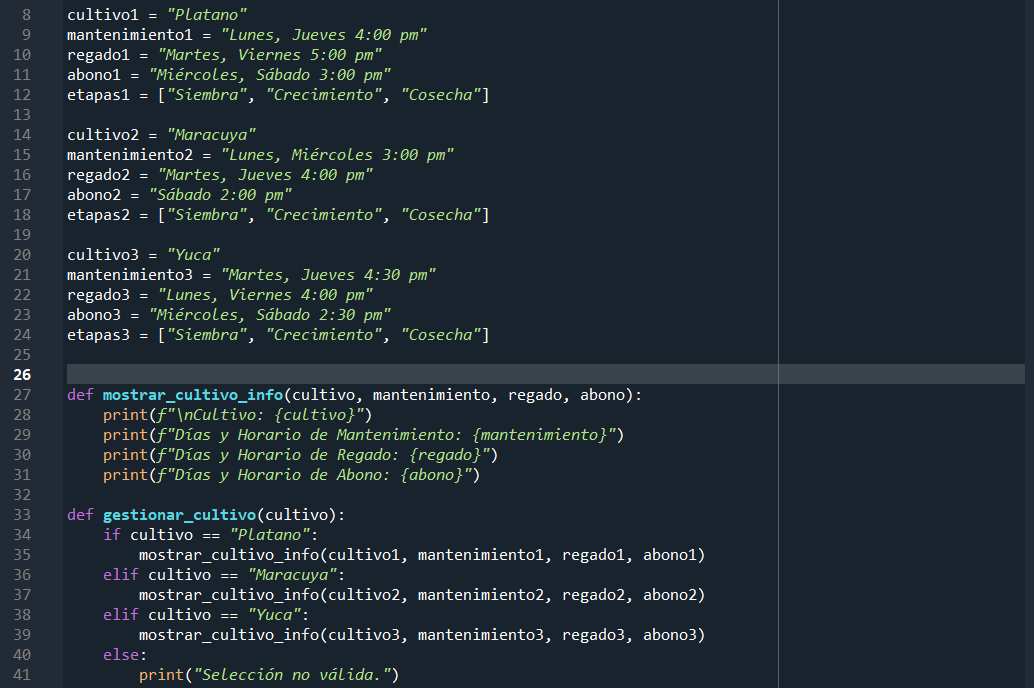
**1 paso:** en lugar de tener todo el código en un solo bloque, elegí definir funciones separadas para cada una de las tareas y menús requeridos. Esto me permitirá reutilizar el código y hará que el programa sea más legible. Por otro lado, elegí no introducir los datos por consola, sino que los definí yo mismo de una vez (esta opción era posible según el enunciado del reto 3).

* Con las funciones mostrar\_cultivo\_info(), gestionar\_cultivo(), calcular\_costos(), calcular\_ganancia() e informe\_economico() me ahorro un montón de líneas de código que realicé en el ejercicio anterior utilizando condicionales y bucles. De esta manera se ve más organizado el programa.

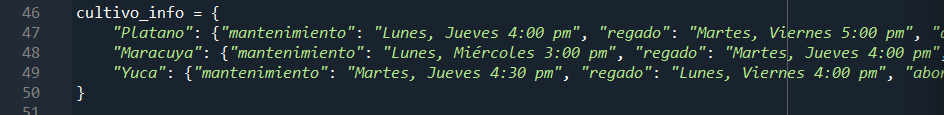


**2 paso:** con lo aprendido en las dos últimas sesiones (estructuras de datos) implementé el uso de diccionarios dado que me permiten meter toda la información de mis cultivos en una sola variable en lugar de muchas variables globales como lo hice inicialmente. De esta manera cada cultivo quedará como las llaves del diccionario y la demás información quedará en forma de valores. Y así solo accedo a las claves del diccionario y sus valores para poder mostrarle al usuario el cultivo que él requiera.

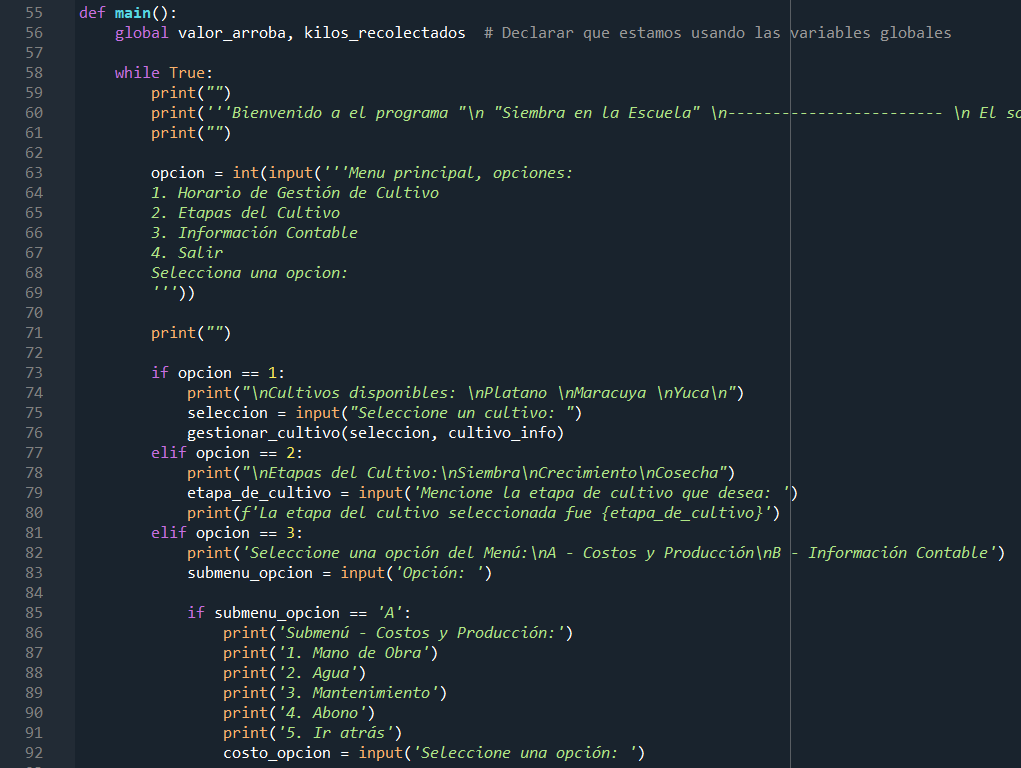
Antes

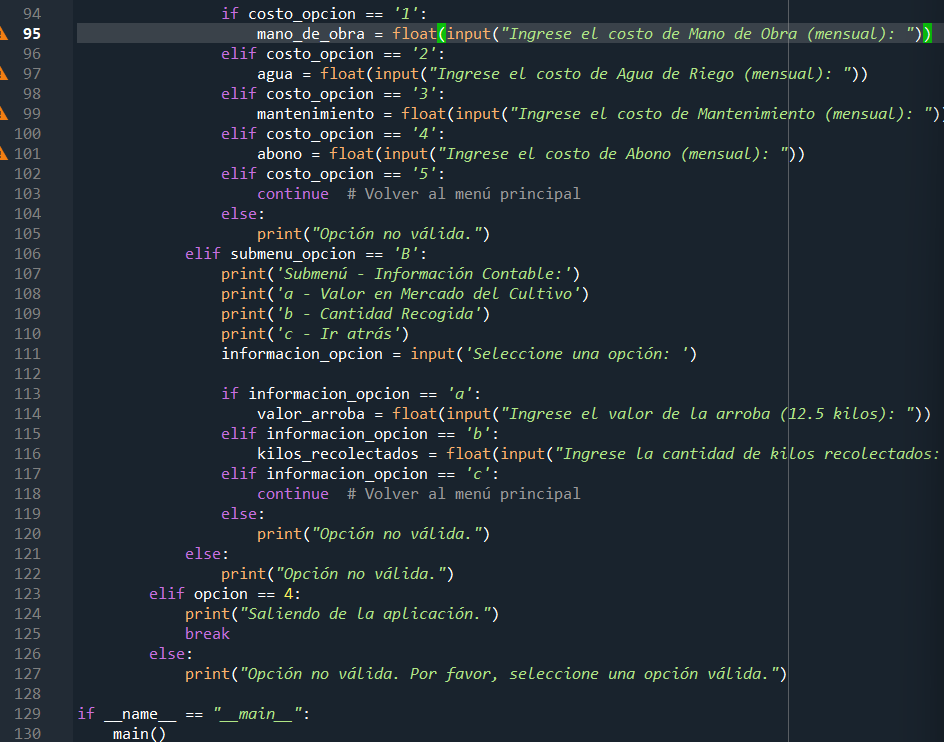


Después



**3 paso:** en este pasó decidí organizar de una mejor manera el menú principal utilizando la función main() y dentro el condicional que tenía originalmente. De igual manera agregué los submenús que pedían en el ejercicio de manera que el usuario pudiera ingresar paso por paso los costos y la producción. Para finalmente dar el informe económico. Es un poco tedioso ingresar los costos uno por uno, pero fue la forma en la que lo pude hacer para cumplir con los requisitos del reto 3, según lo entendí.





Al final de cuentas me pude ahorrar aproximadamente 15 líneas de código además de lograr que mi código se vea un poco más organizado y entendible para otra persona que programe.

Pd: Por favor revisar el código adjunto a este documento (Reto\_3\_improved\_JZ.py) ya que contiene todos los comentarios y cómo se usa cada función. Esos comentarios no quedaron acá en los screenshots para no hacer la imagen más grande. Sin embargo, se pueden encontrar en el código adjunto.

**Punto 2 reto 4**

El código correspondiente a este punto se encuentra adjunto con este Word y se llama (Calculadora\_cientifica.py). Los comentarios y las funcionalidades de cada función se encuentran dentro del mismo código.

PD: el primer punto de este reto se hizo de manera individual puesto que el Reto 3 en mi grupo lo hicimos de manera individual, entonces para darle continuidad al proceso se siguió con la misma dinámica. Para el segundo punto, se hizo en conjunto con mis compañeras Karen Romero y Susana López. Todos hicimos el código por nuestra cuenta y luego lo socializamos, así que puede haber ciertas variaciones.