# ITLA - Instituto Tecnológico de Las AméricasInstituto Tecnológico de Las Américas (ITLA) Departamento de Educación Permanente Fundamentos de Programación con Python Trabajo Final

Nombres y apellidos:

Andres Rodriguez Liberato y Zoar Reyes T.

# Fecha: 14/09/2024

## Trabajo Final: "Gestión de Lista de Tareas" Descripción del Proyecto

El objetivo de este proyecto es crear un programa de consola en Python que permita al usuario gestionar una lista de tareas. El programa debe permitir agregar nuevas tareas, modificar las existentes, marcar tareas como realizadas y mostrar todas las tareas pendientes. Los estudiantes aplicarán conceptos fundamentales como condicionales, bucles, y el uso de diccionarios.

## Requisitos del Proyecto

1. **Menú principal**:

o El programa debe mostrar un menú con las siguientes opciones:

* 1. Agregar Tarea
  2. Modificar Tarea
  3. Marcar Tarea como realizada
  4. Mostrar Lista de Tareas
  5. Salir
* El programa debe continuar ejecutándose hasta que el usuario seleccione la opción "Salir".
* Se deben utilizar condicionales y un bucle while para gestionar la lógica del menú.

## Agregar Tarea:

* Solicitar al usuario el nombre de la tarea.
* Verificar si la tarea ya existe en el diccionario tareas.
* Si la tarea ya existe, mostrar un mensaje indicándolo. Si no existe, solicitar al usuario la descripción de la tarea y agregarla al diccionario tareas.

## Modificar Tarea:

* Permitir al usuario modificar la descripción de una tarea existente.
* Solicitar al usuario el nombre de la tarea que desea modificar.
* Si la tarea existe, solicitar la nueva descripción y actualizarla en el diccionario. Si no existe, mostrar un mensaje indicándolo.

## Marcar Tarea como Realizada:

* Permitir al usuario marcar una tarea como realizada.
* Solicitar al usuario el nombre de la tarea a marcar como realizada.
* Si la tarea existe, eliminarla del diccionario tareas. Si no existe, mostrar un mensaje indicándolo.

## Mostrar Lista de Tareas:

* Mostrar todas las tareas almacenadas en el diccionario tareas.
* Mostrar el nombre de la tarea y su descripción.
* Si no hay tareas en el diccionario, mostrar un mensaje indicando que no existen tareas.

## Salir:

o Finalizar el programa.

## Ejemplo de Funcionamiento

--- Menú Principal ---

1. Agregar Tarea
2. Modificar Tarea
3. Marcar Tarea como realizada
4. Mostrar Lista de Tareas
5. Salir

Ingrese el número de la opción deseada: 1

Nombre de la tarea: Comprar leche

Descripción de la tarea: Comprar leche en el supermercado. Tarea agregada con éxito.

--- Menú Principal ---

1. Agregar Tarea
2. Modificar Tarea
3. Marcar Tarea como realizada
4. Mostrar Lista de Tareas
5. Salir

Ingrese el número de la opción deseada: 4

Lista de Tareas:

Comprar leche: Comprar leche en el supermercado.

**Criterios de Evaluación**

* **Funcionalidad**: El programa debe cumplir con todas las funciones especificadas.
* **Uso correcto de condicionales, bucles, y diccionarios**: El código debe incluir el uso adecuado de estas estructuras.
* **Gestión de errores**: El programa debe manejar situaciones como la existencia de tareas duplicadas o tareas no encontradas.
* **Legibilidad y organización del código**: El código debe estar bien estructurado, con nombres de variables descriptivos y comentarios cuando sea necesario.

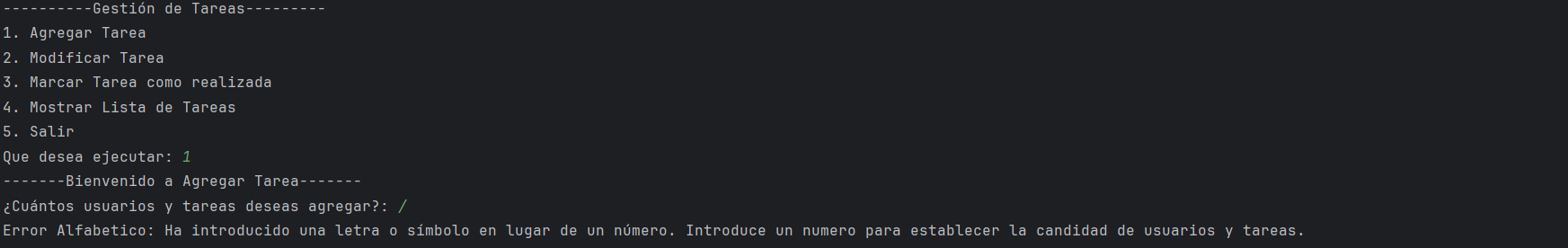
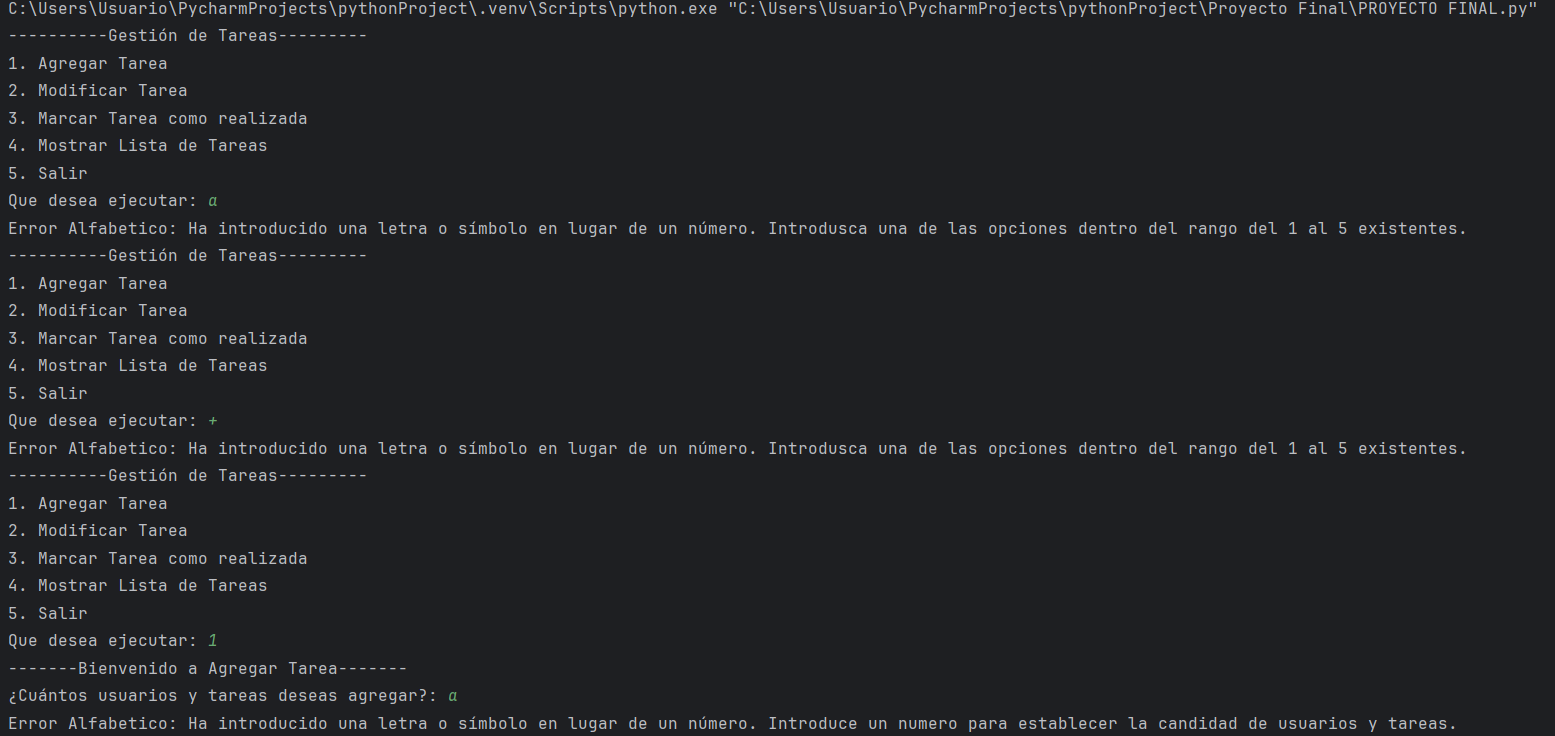
## Modo de entrega

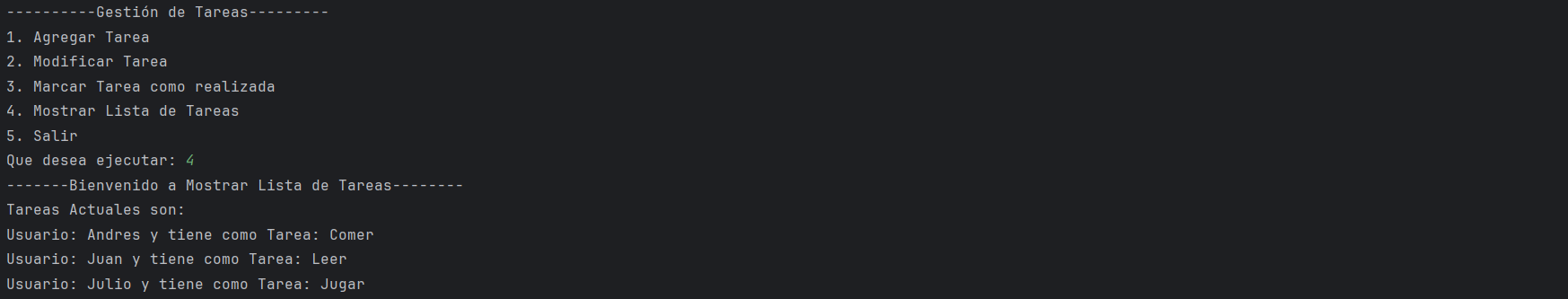
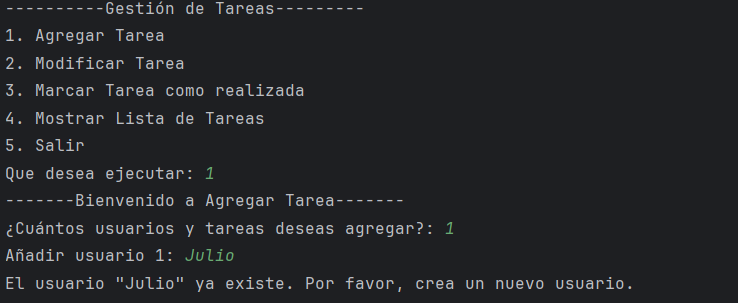
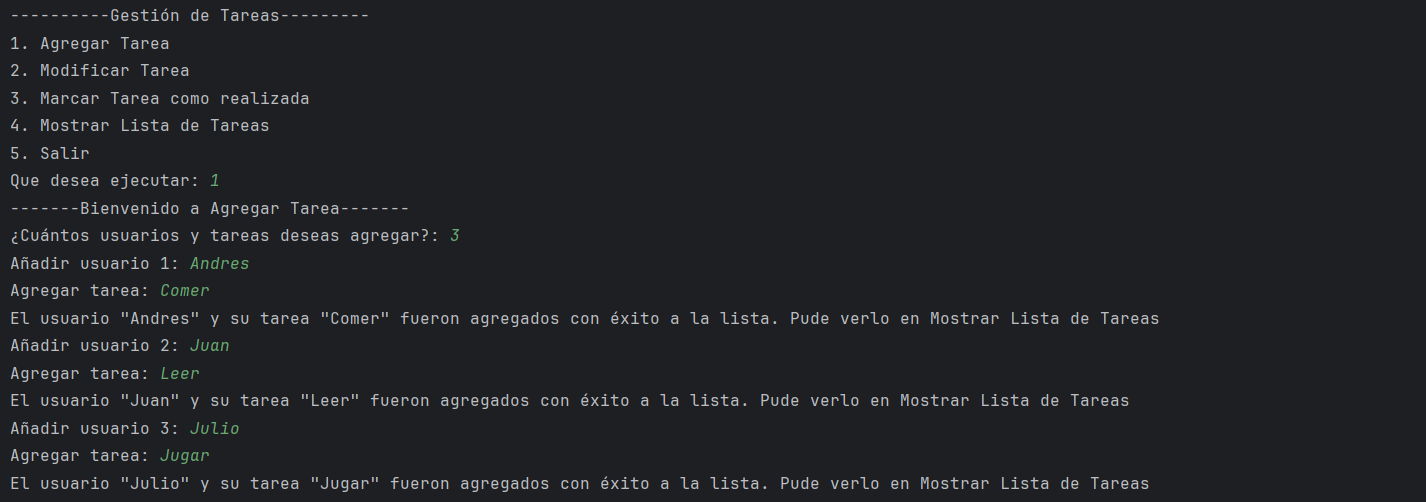
* Copie y pegue el código de su trabajo en un documento de Word, después del código adjunte imágenes de todas las funcionalidades del código funcionando y por último guarden el documento como archivo PDF y súbanlo a la plataforma en el espacio correspondiente al trabajo final.
* Ambos estudiantes deben subir el trabajo a la plataforma

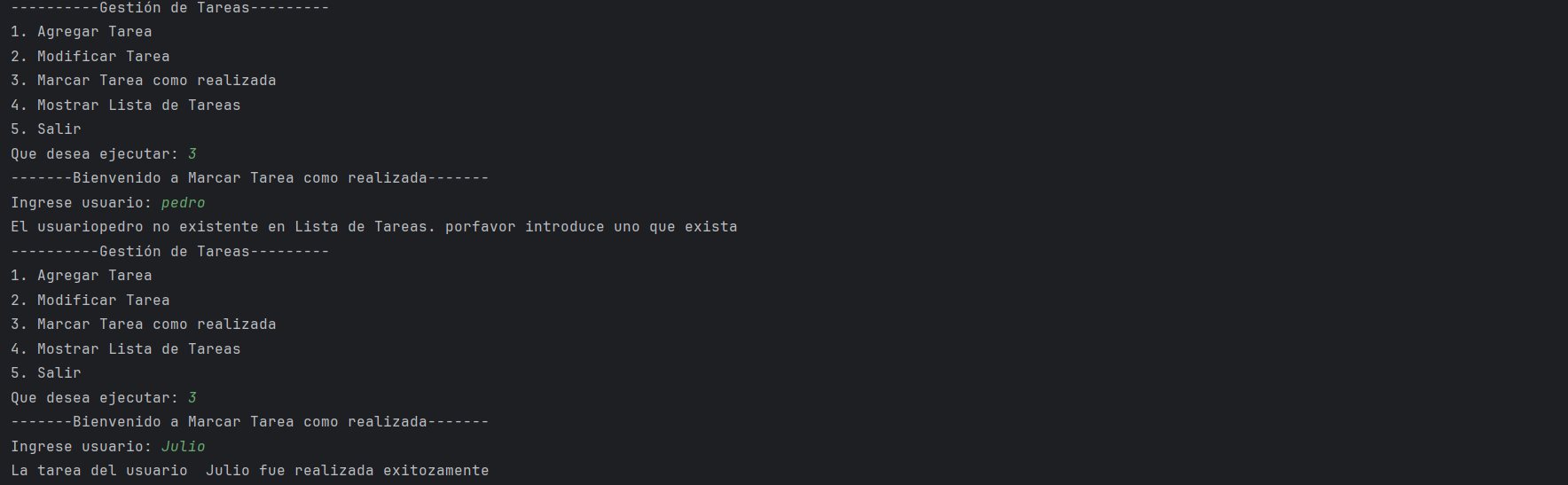
**Código del Programa**

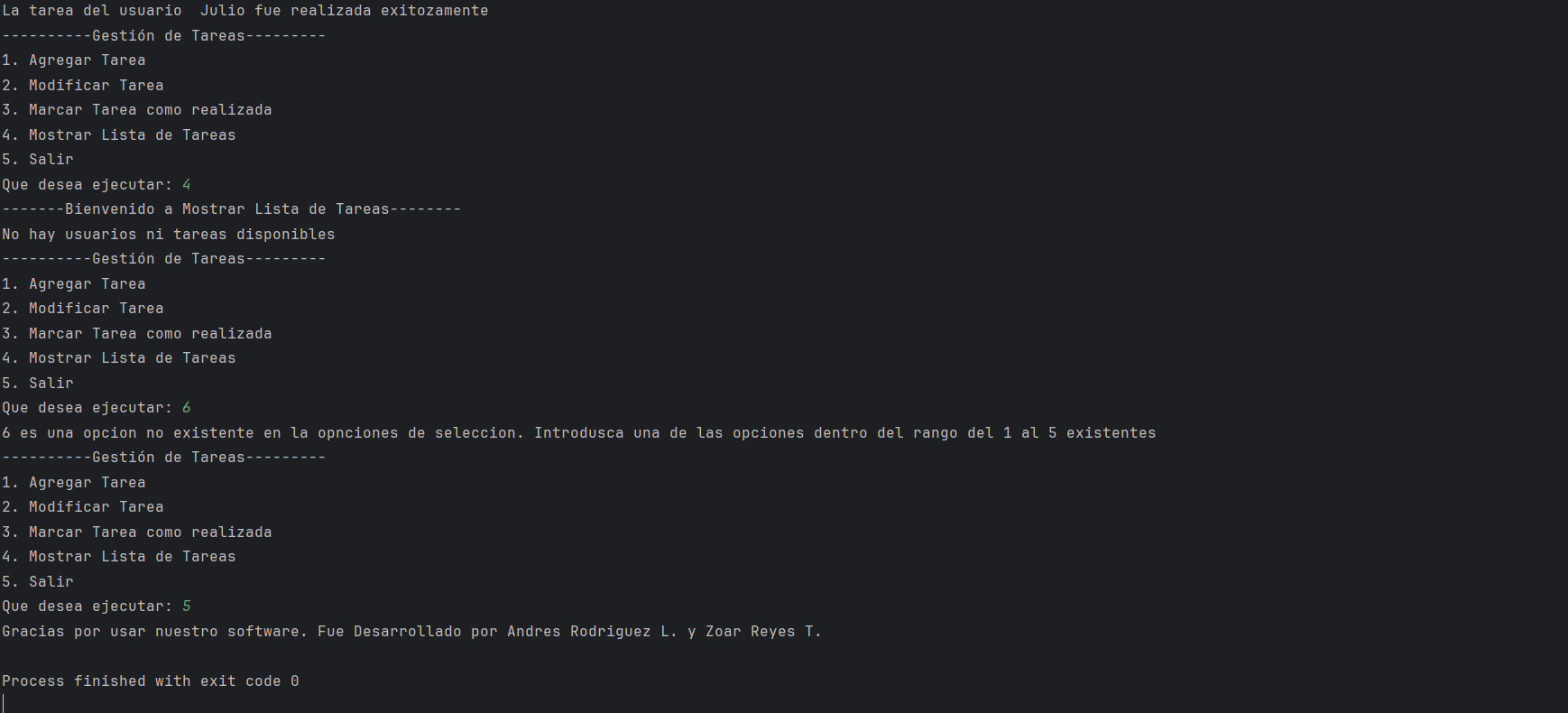
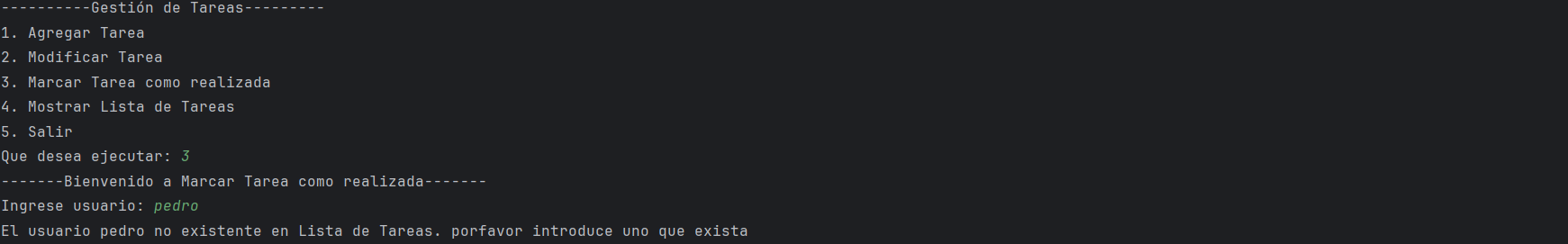
#Practica Final Hecha por Andres Rodriguez Liberato y Zoar Reyes Tolentino  
tareas ={}#Diccionario vacio para luego a{adirle elementos  
def agregarTarea(): #Metodo para añadir nuevos usuarios  
 print("-------Bienvenido a Agregar Tarea-------")  
 try:#Para capturar el error Alfabetico  
 cantidadElementos = int(input("¿Cuántos usuarios y tareas deseas agregar?: "))  
 for cantidadesAgregar in range(cantidadElementos):#Bucle para añadir la cantidad de usuarios y tareas dentro del diccionario tareas  
 anadido = input(f"Añadir usuario {cantidadesAgregar + 1}: ")  
 # Condición para añadir nuevas tareas y claves  
 if anadido not in tareas:  
 claveAnadida = input("Agregar tarea: ")  
 tareas[anadido] = claveAnadida  
 print(f'El usuario "{anadido}" y su tarea "{claveAnadida}" fueron agregados con éxito a la lista. Pude verlo en Mostrar Lista de Tareas')  
 else:#Este mensaje se muestra si el usuario ya existe  
 print(f'El usuario "{anadido}" ya existe. Por favor, crea un nuevo usuario.')  
 except ValueError:#Para delvolver una solucion al error Alfabetico  
 print("Error Alfabetico: Ha introducido una letra o símbolo en lugar de un número. Introduce un numero para establecer la candidad de usuarios y tareas.")  
def modificarTarea(): #Metodo para modificar tarea.  
 print("-------Bienvenido a Modificar Tarea-------")  
 usuario = input("Ingrese usuario: ")  
 if usuario in tareas:# Condicion para modificar las tareas.  
 print("Tarea Existente. Puedes modificarlo")  
 tareaModificada = input("Modificar Tarea: ")  
 tareas[usuario] = tareaModificada  
 print(f'La tarea anterior fue modificada por "{tareaModificada}" exitosamente en la lista. pude verlo en Mostrar Lista de Tareas')  
 else:  
 print(f'{usuario} no existente en Lista de Tareas. porfavor introduce uno que exista')  
def tareaRealizada(): #Metodo para Realizar una Tarea.  
 print("-------Bienvenido a Marcar Tarea como realizada-------")  
 usuarioTarea = input("Ingrese usuario: ")  
 if usuarioTarea in tareas:#Condicion para realizar una tarea.  
 print(f'La tarea del usuario {usuarioTarea} fue realizada exitozamente')  
 del tareas[usuarioTarea]  
 else:  
 print(f'El usuario {usuarioTarea} no existente en Lista de Tareas. porfavor introduce uno que exista')  
  
def mostrarListaTarea(): #Metodo para Mostrar Lista De Tarea.  
 print("-------Bienvenido a Mostrar Lista de Tareas--------")  
 if tareas:#Condicion para Mostrar Lista de Tarea.  
 print("Tareas Actuales son: ")  
 for tarea in tareas:  
 print(f'Usuario: {tarea} y tiene como Tarea: {tareas[tarea]}')  
 else:  
 print("No hay usuarios ni tareas disponibles")  
while True: #Menu Principal para desplegar todas las funciones.  
 print("----------Gestión de Tareas---------")  
 print("1. Agregar Tarea")  
 print("2. Modificar Tarea")  
 print("3. Marcar Tarea como realizada")  
 print("4. Mostrar Lista de Tareas")  
 print("5. Salir")  
 try:#Funciona como forma de capturar un error en el programa  
 seleccion = int(input("Que desea ejecutar: "))  
 if seleccion == 5:  
 print("Gracias por usar nuestro software. Fue Desarrollado por Andres Rodriguez L. y Zoar Reyes T.")  
 break  
 elif seleccion == 1:  
 agregar = agregarTarea()  
 elif seleccion == 2:  
 modificar = modificarTarea()  
 elif seleccion == 3:  
 realizada = tareaRealizada()  
 elif seleccion == 4:  
 mostrar = mostrarListaTarea()  
 else:  
 print(f'{seleccion} es una opcion no existente en la opnciones de seleccion. Introdusca una de las opciones dentro del rango del 1 al 5 existentes')  
 except ValueError:#Funciona como forma de devolver un mensaje para evitar escribir letras o simbolos en el programa  
 print("Error Alfabetico: Ha introducido una letra o símbolo en lugar de un número. Introdusca una de las opciones dentro del rango del 1 al 5 existentes.")

**Ejecución del Programa**



** ** 



****