

Actividad#2

Programación en Python

Presentado por: Luis Charry, Cesar Olascoaga

Tutor: Franklin Doria

Semestre: Noveno

Programa: Ingeniería

Carrera: Ingeniería de Software

Universidad de Cartagena

Cereté

2025

Introducción.....	3
Implementación de la clase y métodos.....	3
Clase Contacto.....	3
Clase Agenda.....	3
Funcionamiento general del programa.....	3
Conclusión.....	4

Introducción

El programa implementa una agenda de contactos utilizando dos clases principales: Contacto y Agenda. A continuación, se explica cómo se implementan estas clases, sus métodos y el funcionamiento general del programa.

Implementación de la clase y métodos

Clase Contacto

- Propósito: Representar un contacto individual con sus datos personales.
- Atributos: nombre, apellido, email y teléfono.
- Método constructor (`__init__`): Inicializa un objeto Contacto con los datos proporcionados al crear la instancia.

Clase Agenda

- Propósito: Gestionar una colección de objetos Contacto.
- Atributo: una lista llamada `contactos` que almacena los objetos Contacto.
- Métodos:
 - `agregar(contacto)`: Añade un objeto Contacto a la lista `contactos`.
 - `eliminar(nombre)`: Busca un contacto por nombre y lo elimina de la lista. Si no encuentra el contacto, imprime un mensaje de error.
 - `buscar(nombre)`: Busca un contacto por nombre y muestra sus datos completos (nombre, apellido, email, teléfono). Si no lo encuentra, imprime un mensaje.
 - `actualizar(nombre, nuevo_nombre, nuevo_apellido, nuevo_email, nuevo_telefono)`: Busca un contacto por nombre y actualiza sus atributos con los nuevos valores proporcionados. Si no encuentra el contacto, imprime un mensaje.

Funcionamiento general del programa

1. Menú de opciones: El programa presenta un menú interactivo en consola con las opciones para agregar, eliminar, buscar, actualizar contactos o salir.

2. Interacción con el usuario: Según la opción seleccionada, el programa solicita los datos necesarios (por ejemplo, nombre, apellido, email, teléfono) para realizar la operación.
3. Gestión de contactos: Utiliza la clase Agenda para manejar la lista de contactos y llamar a los métodos correspondientes según la acción elegida.
4. Ciclo continuo: El menú se muestra repetidamente hasta que el usuario elige salir (opción 5).
5. Mensajes de confirmación o error: Después de cada operación, el programa informa si la acción fue exitosa o si hubo algún problema (por ejemplo, contacto no encontrado).

Conclusión

En conclusión, el programa implementa una agenda simple para gestionar contactos mediante una estructura orientada a objetos, facilitando operaciones básicas como agregar, eliminar, buscar y actualizar contactos a través de un menú interactivo en consola.