

	<p style="text-align: center;"><b>TALLER 2</b></p> <p style="text-align: center;"><b>Taller de Programación en Python</b></p>	<p style="text-align: right;">VERSIÓN 1 GP-FR-012 F. VIGENCIA 21/09/2017</p>
---	---	--

## Instrucciones Generales

- Organiza tus respuestas en un documento de PDF y entrégalo de manera limpia y profesional. O a través de un <https://colab.research.google.com/>
  - NO utilizar inteligencia artificial. Usa tus propias palabras.
- 

## Ejercicio 1: Manipulación de Listas

### Ejercicio 1: Personalizar Mensajes de Bienvenida

- **Descripción:** Crea un programa para personalizar el mensaje de bienvenida según la edad de la persona.
  - **Pistas:**
    - Si la persona es mayor de edad (18+), felicítala.
    - Si es menor de edad, menciona cuántos años faltan para que sea mayor.
- 

### Ejercicio 2: Determinando la Generación

- **Descripción:** Ampliar la funcionalidad del código para determinar a qué generación pertenece la persona (Generación Z, Millennial, etc.).
  - **Pistas:**
    - Añade una función anidada que clasifique la generación según la edad.
    - Incluye esta información en el mensaje final.
- 

### Ejercicio 3: Verificar Condiciones Especiales

- **Descripción:** Agrega una función que verifique si la edad de la persona es un número redondo (como 20, 30, 40, etc.).
- **Pistas:**
  - Si la edad es un número redondo, genera un mensaje especial.
  - Usa el operador módulo (%) para verificar si la edad es divisible por 10.

---

## Ejercicio 4: Lista de Contactos

**Objetivo:** Crear un programa en Python para gestionar una lista de contactos utilizando un diccionario.

### Descripción del Ejercicio:

Vas a desarrollar una aplicación que permita al usuario gestionar una lista de contactos. El programa debe permitir las siguientes acciones:

1. **Agregar Contacto:** Permite al usuario ingresar el nombre y el teléfono de un contacto y lo guarda en un diccionario.
  2. **Buscar Contacto:** Permite al usuario buscar el teléfono de un contacto por su nombre.
  3. **Mostrar Contactos:** Muestra todos los contactos guardados en el diccionario.
  4. **Editar Contacto:** Permite al usuario actualizar el teléfono de un contacto existente.
- 

## Ejercicio 5: Organizando una Fiesta

**Objetivo:** Crear un programa en Python para gestionar una lista de invitados a una fiesta utilizando una lista.

### Descripción del Ejercicio:

Vas a desarrollar una aplicación que permita al usuario gestionar una lista de invitados a una fiesta. El programa debe permitir las siguientes acciones:

1. **Agregar Invitado:** Permite al usuario ingresar el nombre de un invitado y lo añade a la lista de invitados.
2. **Eliminar Invitado:** Permite al usuario eliminar el nombre de un invitado de la lista.
3. **Verificar Invitado:** Permite al usuario verificar si un nombre específico está en la lista de invitados.
4. **Editar Invitado:** Permite al usuario cambiar el nombre de un invitado en la lista.