

# ENTREGA DE YURI URZUA

## Código Python Completo

```
# Año actual fijo
anio_actual = 2025

# Empleado 1
nombre1 = "Luca"
edad1 = 25
puesto1 = "Desarrollador Junior"
anios_restantes1 = 65 - edad1
anio_jubilacion1 = anio_actual + anios_restantes1

print("==== Registro de Empleado 1 ===")
print("Nombre:", nombre1)
print("Edad:", edad1)
print("Puesto:", puesto1)
print("Año estimado de jubilación:", anio_jubilacion1)
print("")

# Empleado 2 (duplicado y modificado)
nombre2 = "María"
edad2 = 40
puesto2 = "Analista de Datos"
anios_restantes2 = 65 - edad2
anio_jubilacion2 = anio_actual + anios_restantes2

print("==== Registro de Empleado 2 ===")
print("Nombre:", nombre2)
print("Edad:", edad2)
print("Puesto:", puesto2)
print("Año estimado de jubilación:", anio_jubilacion2)
```

## Salida esperada al ejecutar:

==== Registro de Empleado 1 ===

Nombre: Luca

Edad: 25

Puesto: Desarrollador Junior

Año estimado de jubilación: 2045

==== Registro de Empleado 2 ===

Nombre: María

Edad: 40

Puesto: Analista de Datos

Año estimado de jubilación: 2050

## Informe Breve

- **Declaración de variables y cálculo:** Se usaron variables simples como nombre1 = "Luca", edad1 = 25 (enteros y strings). El año actual es 2025.  
Cálculo: anios\_restantes1 = 65 - edad1 y anio\_jubilacion1 = anio\_actual + anios\_restantes1 (aritmética básica).
- **Validaciones futuras (conceptuales):** En versiones avanzadas, usar input() para datos del usuario, try-except para errores (ej: edad negativa), if para rangos (edad 18-65), y isalpha() para nombres solo con letras. No se implementa ahora por ser Lección 1.
- **Reflexión:** Esta práctica refuerza variables como contenedores de datos y print para salida, base esencial para sistemas reales de gestión.