

# Fundamentos de Programación

## Evaluación del curso.

- Individual
- Proyectos y prácticas 100%
  - Se suben a Github (Repositorios privados y con invitación)
  - El trabajo se evalúa en base a cumplimiento de requerimientos
  - Posteriormente hay una evaluación personal y puede modificar la nota del trabajo

Contacto:

Tel. 614 153 52 97

[fer\\_manzanares@hotmail.com](mailto:fer_manzanares@hotmail.com)

## Práctica 1.

Desarrollar un programa en lenguaje Java que calcule para su usuario las operaciones aritméticas básicas: suma, resta, multiplicación y división a partir del ingreso de los valores de los dos operandos.

## Práctica 2.

Desarrollar un programa en lenguaje Java que calcule el equivalente en pies de una medición hecha en metros.

## Instrucciones repositorio github

### ...or create a new repository on the command line

```
echo "# serv_web2025" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/fer-manzanares/serv_web2025.git
git push -u origin main
```

## Primer proyecto del curso.

Se desea construir un **Convertor Universal de Unidades** que permita convertir entre distintas magnitudes (longitud, masa, temperatura, volumen, velocidad, entre otras).

El usuario podrá:

- Seleccionar el tipo de magnitud (p. ej. “Longitud”).
  - Elegir la unidad de origen y destino (p. ej. de “Metros” a “Pulgadas”).
  - Ingresar un valor y obtener el resultado de la conversión.
- 

### Requerimientos funcionales

#### 1. Interfaz gráfica principal con:

- Un *ComboBox* o un *TabPanel* para seleccionar la magnitud (Longitud, Masa, Temperatura, Volumen).
- Dos *ComboBox* para elegir la unidad de origen y destino.
- Un campo de texto para ingresar el valor.
- Un botón “Convertir”.
- Una etiqueta o campo no editable donde mostrar el resultado.

#### 2. Conversión correcta entre unidades predefinidas, por ejemplo:

**Longitud:** metros, kilómetros, millas, pulgadas.

**Masa:** kilogramos, libras, gramos.

**Temperatura:** Celsius, Fahrenheit, Kelvin.

**Volumen:** litros, mililitros, galones.

#### 3. Validaciones:

- Solo se aceptan valores numéricos.
  - Manejo de errores (por ejemplo, si no se introduce valor o si hay una unidad no compatible).
-

## Criterios de evaluación

Criterio	Descripción	Ponderación
Diseño modular	Separación clara entre lógica y GUI	10%
Funcionalidad de conversión	Conversión correcta entre unidades, número de unidades y de conversiones	60%
Interfaz gráfica	Diseño limpio, intuitivo y funcional	10%
Validaciones y manejo de errores	Evita fallos ante entradas incorrectas	20%

---

## Extensiones opcionales (Puntos extra)

- Añadir un historial de conversiones.
- Incorporar selector de idioma (Español/Inglés).
- Mostrar fórmulas usadas en cada conversión.
- Implementar tema claro/oscuro.