

¿Qué demonios  
es la  
programación  
funcional?



# Betabeers



**La programación funcional es  
un paradigma que trata la  
computación como la  
evaluación de funciones  
altamente componibles,  
evitando estados y datos  
mutables.**

# Funciones

**int => string** = x => x.toString

**(int, int) => int** = (x, y) => x \* y

**(int, int) => int** = (m, a) => m \* a

# Tipos

# Álgebra de tipos

## producto (combinación)

Tuple(string, boolean)

Point(int, int)

## suma (uno u otro)

Option[String]

List[N]

Either[A, B]

Id[P]

# Constructores de tipos

DatabaseOperation[R]

Futuro[S]

Option[T]

# Type Classes

# Inmutabilidad

```
pureF(x: Int, y: Int) =>  
    x + y
```

```
pureG(s: String) =>  
    s.trim().toUpperCase()
```

# Pureza

```
impureF(x: Int, y: Int) =>  
    x = 7  
    x + y
```

```
impureG(s: String) =>  
    log("input value is " + s)  
    s.trim().toUpperCase()
```

# Transparencia referencial

```
(x: Int, y: Int, z: Int) => {  
    a = sum(x, y)  
    b = sum(y, z)  
    c= sum(x, z)  
    return a * b * c  
}
```

```
(x = 1, y = 2, z = 3) => {  
    a = sum(1, 2)  
    b = sum(2, 3)  
    c= sum(1, 3)  
    return a * b * c  
}
```

```
(x = 1, y = 2, z = 3) => {  
    a = 3  
    b = 5  
    c= 4  
    return a * b * c  
}
```

# High Order Functions

# Currying

$f = \text{exponent: Int} \Rightarrow \text{base: Int} \Rightarrow \text{base}^{\wedge} \text{exponent}$

$f(2) = \text{base: Int} \Rightarrow \text{base}^{\wedge} 2$

$f(2)(3) = 9$

# Recursividad

```
factorial(n: Int, acc: Int) => {  
    if (n == 0) acc  
    else factorial(n - 1, acc * n)  
}
```

# Concurrencia

# Mónadas

# Composición de funciones

**La programación funcional es  
un paradigma que trata la  
computación como la  
evaluación de funciones  
altamente componibles,  
evitando estados y datos  
mutables.**



 Scala

 Kotlin



elixir

 JS



 PURESRIPT

 elm

 Haskell



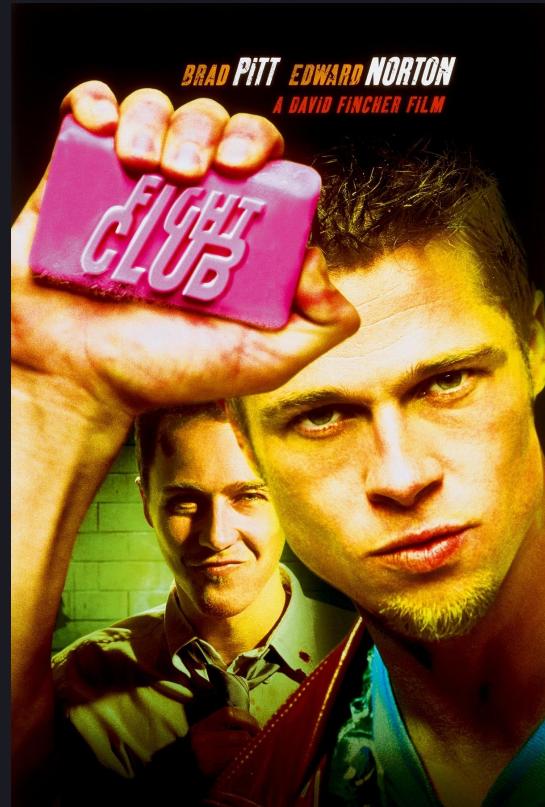
 ERLANG

“

---

## Reglas del club de la programación funcional:

1. No debes dejar de hablar de la programación funcional
2. ¡Nunca dejes de hablar de la programación funcional!



## **Lecturas recomendables:**

- Podcast sobre programación funcional <https://www.ivoox.com/19348205>
- A Pragmatic Introduction to Category Theory - Daniela Sfregola (Lamba World 2018)  
<https://youtu.be/MvOxNm5gn8g>
  - Functores, Aplicativos y Mónadas en imágenes (versión en Español):  
<https://medium.com/@miguelsaddress/funtores-aplicativos-y-m%C3%B3nadas-en-im%C3%A1genes-21aboe6ofe23>