

# Paradigmas

**Paradigma imperativo:** Consiste en explicar muy bien como funciona el código. Las instrucciones deben de ser muy explícitas. El "como" realizamos cada paso del algoritmo debe de ser muy claro. Ejemplos: Fortran, Java, Python, Ruby

**Paradigma declarativo:** Prioriza la claridad del resultado, escribe problemas de manera más corta y precisa, prioriza que funciona con un nivel de abstracción más alto que el imperativo, deja margen para la optimización. Ejemplos: Prolog, LISP, Haskell, R, Regex, SQL

## Declarativo

- \* Utiliza el principio de razonamiento lógico
- \* Se enfoca en el qué es la solución al problema
- \* No necesita definir algoritmos
- \* Deja margen para la optimización
- \* Se centran en hechos y reglas
- \* Alto nivel de abstracción
- \* Simplifica el desarrollo

## Imperativo

- \* Se basa en el cómo
- \* Especificación paso a paso
- \* Problemas de sincronización
- \* Procedimientos y funciones
- \* Control de flujo
- \* Secuencia de instrucciones de falladas
- \* Problemas de sincronización

# Paradigmas

Declarativo		Imperativo
- Funcional		- POE - POO
- LISP		- Visual Basic
- Haskell		- C++
- Python		- Java
- Kotlin		- C#
- Scala		- ADA
- Lógico		- Estructurada
- Prolog		- C
- Mercury		- Ruby
- Scala		- PHP
- Assembly		- Python
		- Fortran
		- P. Aspectos
		- Spring
		- Aspect J