

Capítulo 1. Introducción a la programación funcional

¿Qué es la programación funcional?

Paradigma de programación que considera el cómputo como la evaluación de funciones matemáticas que evitan los cambios de estado y los datos mutables.

Paradigmas de programación

Imperativos

Estructurado o Procedural

- C
- Pascal
- COBOL

Orientado a Objetos

- C++
- Java
- Python
- Ruby

Declarativos

SQL

-SELECT * FROM mi_tabla WHERE ap_paterno = "Amaya"

Funcional

- LISP
- Haskell
- Clojure
- Erlang/Elixir

Funciones matemáticas

Paradigma de programación que considera el cómputo como la evaluación de funciones matemáticas que evitan los cambios de estado y los datos mutables.

* $y = mx + b$

Función pura

Tiene argumentos

NO producen efectos secundarios

Retorna siempre el mismo valor con la misma entrada

Ejemplo:

- Math.sqrt(49) -> 7
- Math.sqrt(49) -> 7

Función Impura

No tiene argumentos

Puede retornar valores distintos

Ejemplo:

-Math.random() -> 0.05

-Math.random() -> 0.93

Programación Funcional

Paradigma de programación que considera el cómputo como la evaluación de funciones matemáticas que evitan los cambios de estado y los datos mutables.

*Evitar los cambios de estado evitando el uso de variables.

*Uso de variables inmutables.

*Reduce la cantidad de errores (bugs).

Mutabilidad

-Variables u objetos pueden ser modificados después de haber sido creados.

Inmutabilidad

-El estado de las variables u objetos no puede ser modificados una vez creados.

POO vs PF

“It is better to have 100 functions operate on one data structure than 10 functions on 10 data structures”. Alan Perlis

Programación Orientada a Objetos:

-Más estructuras de datos.

-Pocas funciones por estructura de datos.

-10 funciones para 10 estructuras de datos.

Programación Funcional:

-Pocas estructuras de datos.

-Más funciones por estructura de datos.

-100 funciones para 1 estructura de datos.