

PARADIGMAS DE PROGRAMACION	DESCRIPCION	EJEMPLOS	FUNCION
• Programación imperativa	Es un estilo de programación que se basa en llamar funciones. Su filosofía es dividir por pasos para completar una tarea. Cualquier proceso puede ser llamado durante la ejecución de un programa.	, C, Pascal, Basic, COBOL  printf( "Hola mundo." ); getch(); /* Pausa */return 0;	Sirven para la estructuración de datos.
• Programación funcional:	Es una programación centrada en las funciones, todos los elementos pueden entenderse como funciones, por lo que pueden ejecutarse mediante llamados de funciones secuenciales.	Ruby, Visual Basic, ML, Scala.  puts "curso: ruby on rails. nivel básico".size	Estos programas ante todo se usan para crear webs.
• Programación Declarativa	Es un lenguaje basado más en las matemáticas y en la lógica que los lenguajes imperativos, por lo que es más cercano al razonamiento humano	Prolog, Lisp, Haske  padrede('Juan', 'María'). <i>% Juan es padre de María</i>	Sirven para generalizar expresiones a través de la introducción de variables.

• Programación POO – Orientada a Objeto	Esta programación se basa en el concepto clases y objetos. Este tipo de programación se usa para estructurar un programa de software en piezas simples y reutilizables	, Java, C#, Python  i = 1  while i <= 25:  print(i),  i += 1	Sirven para construir aplicaciones web, analizar datos, automatizar operaciones y crear aplicaciones empresariales fiables y escalables.
---	--	--	--

• Programación procedimental:	Es un paradigma de la programación, consiste en basarse de un número muy bajo de expresiones repetidas, englobarlas todas en un procedimiento o función	Miranda, Erlang, Kotlin  1> c(cesar). {ok, cesar}2> cesar:cifrar("hola"). 'u yn'3> cesar:descifrar("u yn"). hola	Sirven para hacer aplicaciones que puedan soportar fallos.
• Programación lógica	Es un paradigma que establece problemas mediante sentencias, lo que obliga a tener un alto nivel de abstracción	Mercury, Logtalk, Curry.  ?- write('Hello world'), nl. Hello world true. Logtalk - <a href="https://es.qaz.wiki/wiki/Logtalk">https://es.qaz.wiki/wiki/Logtalk</a>	proporciona soporte para encapsulación y ocultación de datos , separación de preocupaciones y reutilización mejorada de código

[https://es.wikipedia.org/wiki/Programaci3n\\_por\\_procedimientos](https://es.wikipedia.org/wiki/Programaci3n_por_procedimientos) <https://profile.es/blog/que-es-la-programacion-orientada-a-objetos/> <https://www.ionos.es/digitalguide/paginas-web/desarrolloweb/programacion-funcional/> <https://aprenderly.com/doc/2418556/programaci3n-procedimenta>

