

Historia de la programación

1. Inicio: Década de los 40, con ENIAC, se usaban secuencias de código base, tomando el sistema de palabras, entre 50 y 70 con fortran, lisp y cobol se siguen utilizando, también salió bspl (antecesor a C)
2. Década de los 70: uso de c y pascal, y el origen de sistemas de uso múltiple.
3. Década de los 80: Inicio de lenguajes de nicho, ADA (defensa), perl (ficheros de texto), perl (de donde sale tk)
4. Década de los 90: Lenguajes de propósito múltiple, cosas como java (uso de sistemas embebidos), y javascript (aplicaciones web), python (prototipos, ml, backend) y php.
5. Décadas posteriores (actionscript con aplicaciones flash) C# (financiero) go (backend) clang y clojure, swift con Apple.

Tipos de lenguajes de programación

1. Lenguajes compilados: Aquellos que desde el código se genera un sistema que pueda leer directamente el sistema operativo. Tipos: C, go, ensamblador. Ventajas:

- Más rápido: se manda directo con respecto al procesador.
-

Desventajas:

- No funciona de la misma manera para todas las arquitecturas de procesadores

2. Lenguajes interpretados: Aquellos que requieren de una herramienta que se llama bytecode, mandando llamar un intérprete que lee el código y lo ejecuta sobre sí mismo, siendo más lento que los compilados. Tipos: java (necesita una java virtual machine), python, perl, php. Ventajas:

- No requiere hacer ajustes para el procesador

Desventajas:

- más lento

3. Lenguajes híbridos: Usa un intérprete denominado JIT (just in time)

Otra manera de clasificarlos es mediante los tipados o no tipados:

1. Lenguajes tipados: Se indica desde el principio la variable que será
2. Lenguajes no tipados: El intérprete lee y explica los tipos de variables que se almacenen.

Los tipos de aplicaciones pueden ser:

1. escritorio
2. web
3. móvil