

Informe de Lenguajes, Paradigmas y Estándares de Programación

Los lenguajes de programación son herramientas que permiten crear software para realizar diversas tareas y ejecutarlas en diferentes dispositivos. Existen muchos tipos de lenguajes de programación, cada uno con su propio objetivo y enfoque.

Con ellos, podemos desarrollar procedimientos, reglas, automatizaciones y mucho más. Los paradigmas de programación son estilos usados para resolver problemas específicos dentro de la programación. (concepto, 2023)

Algunos de los paradigmas más conocidos son la programación orientada a objetos, la programación funcional y la programación imperativa.

Ya que los lenguajes de programación suelen ofrecer mucha libertad a la hora de escribir el código, se crearon los estándares de programación. Estos tienen el propósito de unificar la estructura, el orden y la forma de escribir código según el lenguaje de programación.

Es importante seguir los estándares ya que nuestro código puede ser leído y modificado por otras personas, y debemos facilitarles la comprensión y el mantenimiento de este. (rramirez, 2023)

Los lenguajes de programación pueden clasificarse en tres niveles:

- *Bajo*
- *Medio*
- *Alto*

Cuanto más 'bajo' sea el lenguaje, más cerca está del lenguaje de la máquina en la que es ejecutada, por lo general suelen ser complejos de leer y escribir ya que se alejan del lenguaje humano, y los altos son todo lo contrario.

Según la tarea a realizar, se debe elegir correctamente el lenguaje de programación, de lo contrario, se corre el riesgo de enfrentar dificultades en el desarrollo y mantenimiento de este, incluso ser ineficientes. (rockcontent, s.f.)

Algunos de los lenguajes de programación de **bajo** nivel a continuación, estos tienen la característica de ejecutarse de una forma más eficiente y rápida:

➔ Lenguaje ensamblador y Lenguaje máquina

```
-u 100 1a
OCFD:0100 BA0B01 MOV DX,010B
OCFD:0103 B409 MOV AH,09
OCFD:0105 CD21 INT 21
OCFD:0107 B400 MOV AH,00
OCFD:0109 CD21 INT 21
-d 10b 13f
OCFD:0100 20 65 73 74 65 20 65 73-20 75 6E 20 70 72 6F 67 Hola,
OCFD:0110 72 61 6D 61 20 68 65 63-68 6F 20 65 6E 20 61 73 este es un prog
OCFD:0120 73 65 6D 62 6C 65 72 20-70 61 72 61 20 6C 61 20 rama hecho en as
OCFD:0130 57 69 6B 69 70 65 64 69-61 24 sembler para la
OCFD:0140
```

Ilustración 1 Lenguaje Máquina - máquina Intel 8088 (https://es.wikipedia.org/wiki/Lenguaje_ensamblador)

En el nivel **medio** como su nombre indica, es un punto intermedio entre los altos y bajos, estos suelen usarse para programar sistemas y aplicaciones de bajo nivel, entre ellos:

➔ C y C++

```
#include <stdio.h>

int main(void)
{
    printf("Hello World\n");
    return 0; // ends the program
}
```

Ilustración 2 Lenguaje C – (<https://medium.com/streamelopers/hello-world-en-10-lenguajes-a30f73d771c4>)

Por último, tenemos los lenguajes de nivel **alto**, los cuales son más cercanos al lenguaje humano por lo que son más fáciles de leer y escribir, se usan para crear aplicaciones de alto nivel y creación de software, algunos de ellos son:

➔ Python y JavaScript

```
print('Hello ,world');
```

Ilustración 3 Lenguaje Python - (<https://medium.com/streamelopers/hello-world-en-10-lenguajes-a30f73d771c4>)

Los paradigmas de programación son utilizados para resolver problemas específicos, algunos de los más significantes:

- **Paradigma Imperativo:** Centrado en explicar cómo funciona el código, cada acción dentro de nuestro código debe quedar plasmado de una manera explícita.
- **Paradigma Declarativo:** Contrario al primero, este prioriza el resultado por encima del 'paso a paso'.
- **Paradigma Orientado a Objetos:** Los datos son almacenados en forma de campos con atributos y propiedades, el código se almacena en forma de procedimientos.
- **Paradigma Funcional:** Las funciones se pueden pasar como argumentos, devolver como argumentos o almacenar en estructuras de datos.
- **Paradigma Lógico:** Basada en la lógica matemática, los objetivos los expresa como una colección de afirmaciones acerca de los resultados. (platzi, 2023)

La importancia de los estándares de programación viene de todos los problemas que supondría no seguirlas, ya que de no hacerlo, los programadores no podrían trabajar juntos de manera efectiva, por lo que no serían capaces de crear código de alta calidad, supondría un tiempo extra en acabar proyectos por lo que se perdería dinero a largo plazo además de ser difícil de actualizar y mantener. (rramirez, 2023)

- **PEP 8 (Python Enhancement Proposal 8):** Uso de espacios en lugar de tabulaciones. (python, 2023)
- **Google Java Style:** Uso de mayúsculas y minúsculas en nombres de variables. (github, 2023)

Para concluir, los lenguajes, paradigmas y estándares de programación son herramientas esenciales en el desarrollo de software. Cada uno tiene ventajas y desventajas, y la elección adecuada depende del problema a resolver. Es crucial comprender los fundamentos de estos elementos para escribir código eficiente y elegante. Los lenguajes son la interfaz entre el programador y la máquina, con diferentes características que los hacen adecuados para distintos problemas. Los paradigmas influyen en la organización y ejecución del código, y cada uno tiene su filosofía. Los estándares de programación son reglas que garantizan la consistencia y claridad del código. Seguirlos mejora la legibilidad y mantenibilidad del código.

Bibliografía

concepto. (10 de 11 de 2023). Obtenido de <https://concepto.de/lenguaje-de-programacion/>

github. (10 de 11 de 2023). Obtenido de <https://google.github.io/styleguide/javaguide.html>

platzi. (10 de 11 de 2023). Obtenido de <https://platzi.com/blog/paradigmas-programacion/>

python. (10 de 11 de 2023). Obtenido de <https://peps.python.org/pep-0008/>

rockcontent. (s.f.). Obtenido de rockcontent.com

rramirez. (10 de 11 de 2023). Obtenido de <https://web.rramirez.com/estandares-basicos-de-programacion/>