

Herramientas digitales del Pensamiento Computacional

Las herramientas digitales del pensamiento computacional son recursos tecnológicos que apoyan el desarrollo de habilidades clave como la descomposición, el reconocimiento de patrones, la abstracción y el diseño de algoritmos. Estas herramientas permiten a estudiantes y profesionales resolver problemas de forma lógica y estructurada. A continuación, se presenta una lista organizada por tipo de herramienta:

1. Herramientas de programación visual (bloques)

Estas herramientas son ideales para introducir conceptos básicos de programación y pensamiento lógico.

- Scratch (MIT): Programación por bloques, muy usada en educación básica.
- Blockly (Google): Similar a Scratch, usado en aplicaciones como Code.org.
- App Inventor (MIT): Para crear apps móviles con bloques.

2. Lenguajes de programación (texto)

Usados para desarrollar habilidades de codificación más avanzadas.

- Python: Simple y potente, ideal para aprender a programar y resolver problemas complejos.
- JavaScript: Muy usado para desarrollo web.
- C/C++ y Java: Usados en contextos más técnicos y estructurados.

3. Plataformas de aprendizaje interactivo

Estas plataformas enseñan pensamiento computacional de forma gamificada o guiada.

- Code.org: Actividades interactivas para todos los niveles.
- Tynker: Similar a Code.org, muy usado en escuelas.
- Kodable y Lightbot: Juegos educativos para niños que introducen lógica de programación.
- CS First de Google: Currículo gratuito con Scratch.

4. Robots educativos y kits físicos

Fomentan el pensamiento computacional a través de la programación de hardware.

- LEGO Mindstorms y LEGO Spike: Programables con bloques o texto.
- Arduino: Plataforma de hardware abierto que permite enseñar lógica y electrónica.
- Micro: bit: Microcontrolador diseñado para la enseñanza de la programación.

5. Simuladores y entornos virtuales

Permiten experimentar con conceptos abstractos sin necesidad de hardware físico.

- Tinkercad (simulador de circuitos y 3D).
- Algodots: Para entender algoritmos de manera visual.
- NetLogo: Simulación de fenómenos complejos.

6. Herramientas de diseño algorítmico y pseudocódigo

Ayudan a pensar en términos de pasos lógicos antes de codificar.

- PSeInt: Para escribir y simular pseudocódigo en español.
- Flowgorithm: Para crear diagramas de flujo ejecutables.

7. Hoja de cálculo y lógica

Aunque no son específicas de programación, fomentan habilidades clave.

- Microsoft Excel o Google Sheets: Promueven la lógica, funciones, y manejo de datos.
- Khan Academy – Computer Science: Introducción visual y conceptual al pensamiento computacional.