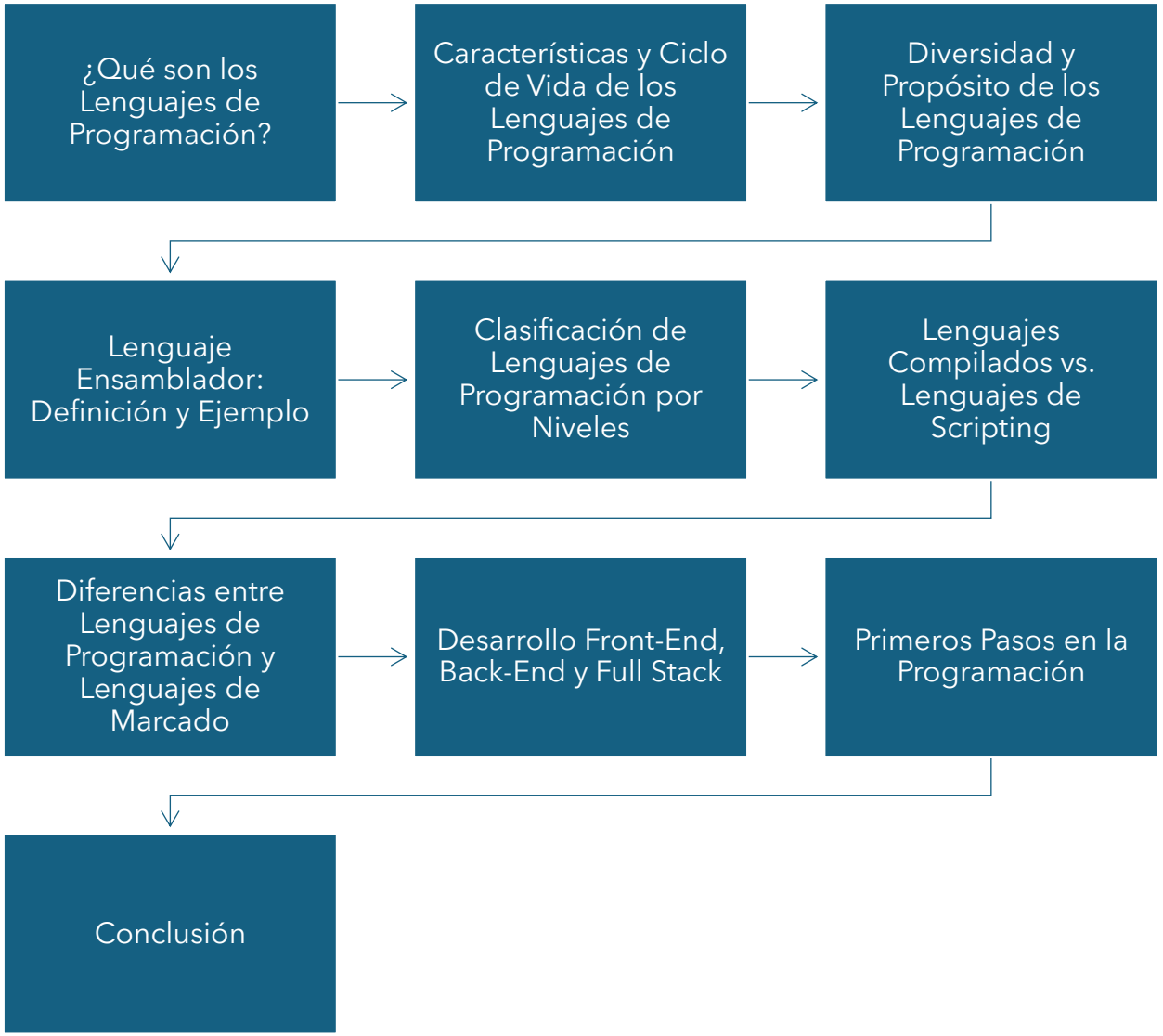
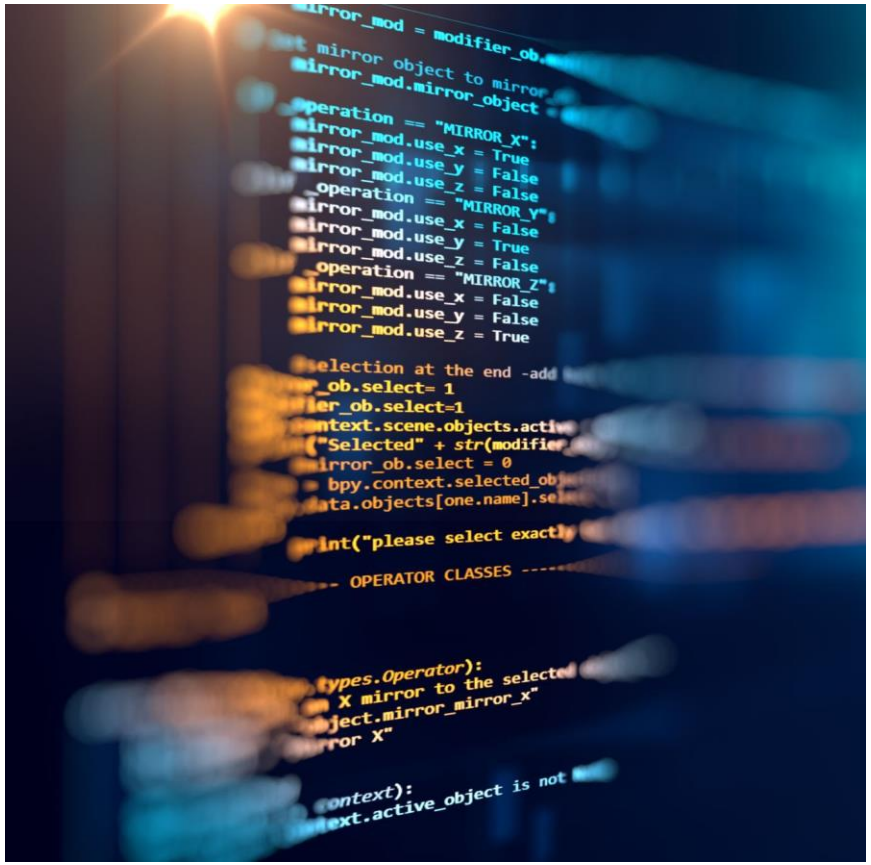
Conceptos básicos de Programación

Contenido


¿Qué son los Lenguajes de Programación?
Características y Ciclo de Vida de los Lenguajes de Programación
Diversidad y Propósito de los Lenguajes de Programación
Lenguaje Ensamblador: Definición y Ejemplo
Clasificación de Lenguajes de Programación por Niveles
Lenguajes Compilados vs. Lenguajes de Scripting
Diferencias entre Lenguajes de Programación y Lenguajes de Marcado
Desarrollo Front-End, Back-End y Full Stack
Primeros Pasos en la Programación
Conclusión

¿Qué son los Lenguajes de Programación?

- Formas Variadas de los Lenguajes de Programación
  - Escritos en distintos lenguajes, similares a los idiomas humanos.
- Características Comunes
  - Universalidad en funcionalidad y equivalencias en sintaxis y estructura.
- Ciclo de Vida de los Lenguajes
  - Desarrollo por innovación y adopción por programadores.
  - Potencial desuso y desaparición si no es ampliamente adoptado.



Características y Ciclo de Vida de los Lenguajes de Programación

Creación de Lenguajes de Programación

- Pueden ser creados por una persona en poco tiempo

Uso del Inglés en Programación

- Palabras clave y sintaxis mayormente en Inglés
- Existen en otros idiomas pero son menos populares

Diversidad y Propósito de los Lenguajes de Programación

Variedad de Lenguajes de Programación

- Cada lenguaje tiene un propósito específico
- Nuevos lenguajes se crean para resolver problemas diversos

Impacto en el Desarrollo de Aplicaciones

- La elección del lenguaje no afecta el desarrollo final
- Las computadoras operan en lenguaje máquina (0s y 1s)

Simplificación de la Programación

- Los lenguajes de programación facilitan el proceso
- Permiten evitar el uso directo de código máquina

Lenguaje Ensamblador: Definición y Ejemplo

Comandos Cortos y Memorables	• Por ejemplo, JMP, MOV y ADD
Atajo para Lenguaje Máquina	• Código fuente más corto y manejable
Traducción a Lenguaje Máquina	• Mediante el uso de ensambladores
Desventajas	• Genera código más grande y lento • Requiere más espacio en disco y memoria • Dificultad de portabilidad entre computadoras

Hola Mundo en Ensamblador

```
01  DATA SEGMENT
02        MESSAGE DB "HELLO WORLD!!!$"
03  ENDS
04
05  CODE SEGMENT
06       ASSUME DS:DATA CS:CODE
07  START:
08          MOV AX,DATA
09          MOV DS,AX
10          LEA DX,MESSAGE
11          MOV AH,9
12          INT 21H
13          MOV AH,4CH
14          INT 21H
15  ENDS
16  END START
17
```

Clasificación de Lenguajes de Programación por Niveles

Lenguajes de Bajo Nivel

- Interacción directa con la CPU
- Ejecutan comandos básicos
- Difíciles de leer
- Ejemplo: Código de máquina con 0s y 1s

Lenguajes de Medio Nivel

- Combinan características de bajo y alto nivel
- Ejemplo: Lenguaje C con uso de apuntadores

Lenguajes de Alto Nivel

- Se asemejan al lenguaje humano
- Fáciles de leer y escribir
- Requieren intérprete o compilador

Lenguajes Compilados vs. Lenguajes de Scripting

Portabilidad de Lenguajes Interpretados

Considerados más portátiles que los compilados

La velocidad de procesadores reduce la brecha de rendimiento

Ejecución de Lenguajes Compilados

Mayor rapidez en la ejecución comparados con interpretados

Uso de Intérpretes

JavaScript, Python y Ruby se ejecutan línea a línea

Función de Compiladores

C++, COBOL y Visual Basic crean archivos ejecutables

Software como Windows o Mac OS X usa lenguajes compilados

Diferencias entre Lenguajes de Programación y Lenguajes de Marcado

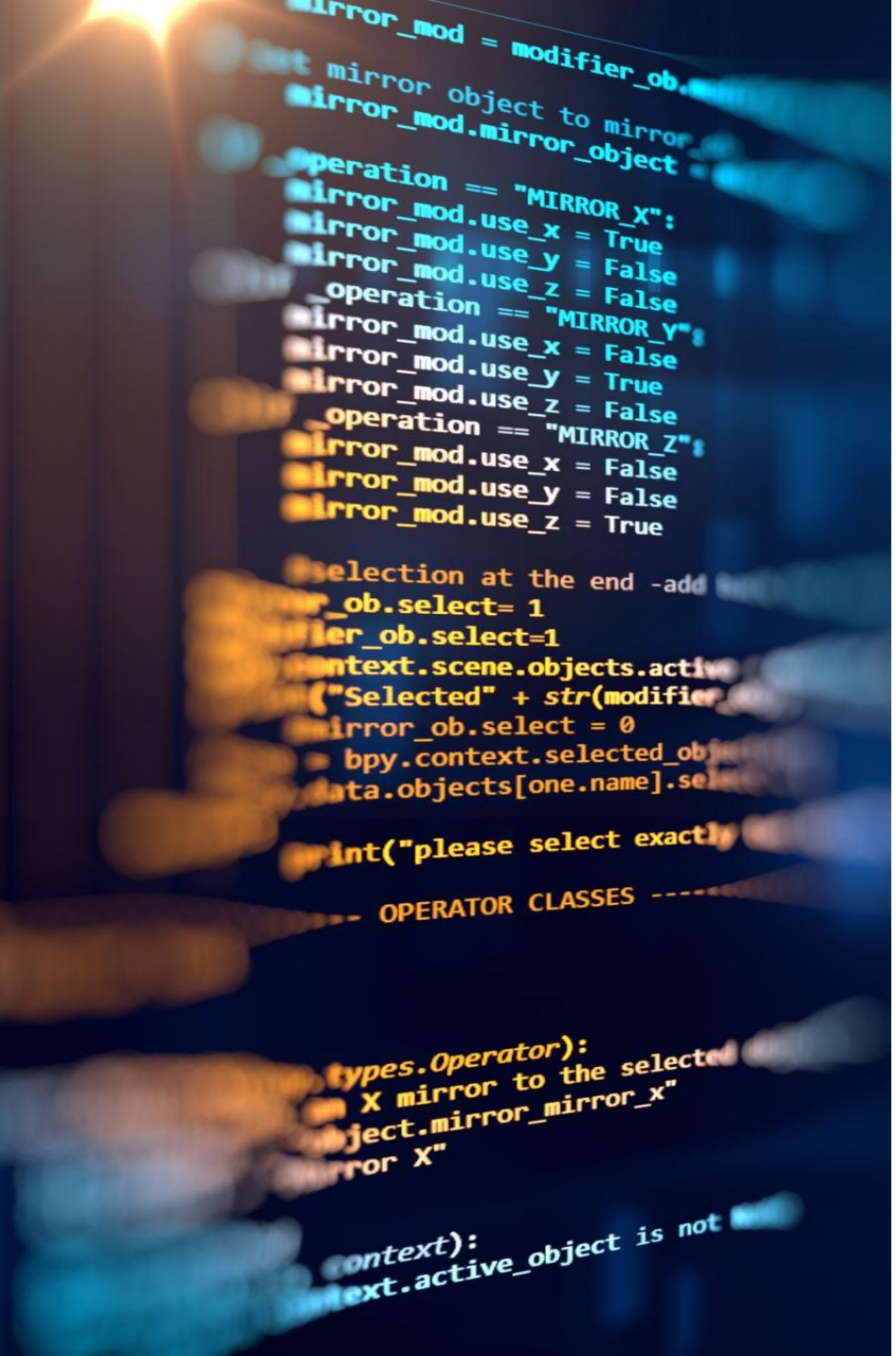
- Diferencias Clave
  - Los lenguajes de marcado estructuran datos, no son para programar.
  - HTML es un ejemplo de lenguaje de marcado.
- Ejemplo de HTML
  - Etiquetas para definir títulos y listas.
  - El código muestra la estructura básica de una página.

```
<html>
    <body>
    <h1>This is heading 1</h1>
    </body>
</html>


\begin{document}
    \setcounter{page}{128}
    \tableofcontents\thispagestyle{empty}
    \pagebreak

    \begin{landscape}
        \pagebreak
        \thispagestyle{portada}
        \hspace{0pt}
        \vfill
        \section{Acta}\label{Anexo1}
        % \HUGE \section{Acta}\label{Anexo1}
        \vfill
        \hspace{0pt}
        \pagebreak
    \end{landscape}

\end{document}
```



Lenguajes de Programación vs. Lenguajes de Marcado

- Lenguajes de Programación
  - Codifican programas y algoritmos.
  - Permiten definir lógica, realizar cálculos y manipular datos.
- Lenguajes de Marcado
  - Estructuran y dan formato a datos, sin definir lógica ni algoritmos.

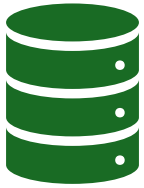
Desarrollo Front-End, Back-End y Full Stack

Desarrollo Front-End

Responsable de la interfaz de usuario (UI)

Incluye elementos visibles como imágenes y botones

Lenguajes como HTML, CSS, JavaScript



Desarrollo Back-End

Maneja aspectos invisibles para el usuario

Compuesto por servidores, bases de datos, APIs

Lenguajes como Java, Ruby, Python, PHP

Desarrollador Full Stack

Capaz de trabajar en front-end y back-end

Primeros Pasos en la Programación

La esencia de la programación

- Compuesta por declaraciones detalladas
- Cada declaración ejecuta una acción específica

Importancia de la precisión

- Los comandos deben ser extremadamente detallados
- La ejecución rigurosa es clave para la eficacia

Desarrollo de programas eficaces

- Comandos detallados solucionan problemas específicos
- El código debe estar bien redactado y estructurado

Conclusión

Composición del Código de Programación

- Secuencia de declaraciones que dirigen acciones específicas
- Comandos ejecutados con riguroso detalle

Desarrollo de Programas Eficaces

- Importancia de redactar comandos detallados
- Resolución de problemas a través de código bien diseñado